

3100000011515



PENGONTROL SISTEM KEMUDI 4 RODA PADA MOBIL DENGAN MENGGUNAKAN ADAPTIF FUZZY LOGIC CONTROLLER

TUGAS AKHIR

PERPUSTAKAAN IT 85-2-00	
Tgl. Terima	17
Terima Dari	8911
No. Agenda Prp.	

Disusun oleh :

DORIS ERWANTORO

NRP. 2292.100.081

RSE

629.89

Erw

P-1

1998



JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1998

PENGONTROL SISTEM KEMUDI 4 RODA PADA MOBIL DENGAN MENGGUNAKAN ADAPTIF FUZZY LOGIC CONTROLLER

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik Elektro**

Pada

Bidang Studi Elektronika

Jurusan Teknik Elektro

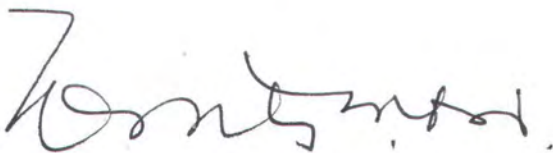
Fakultas Teknologi Industri

Institut Teknologi Sepuluh Nopember

S u r a b a y a

Mengetahui / Menyetujui

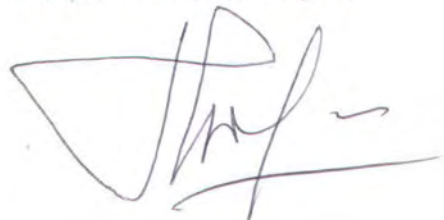
Dosen Pembimbing I



Ir. NAWANTOWIBOWO H.

NIP. 130 368 612

Dosen Pembimbing II



PUJIONO, ST

NIP. 132 094 793

S U R A B A Y A

Agustus, 1998

Abstrak

Sistem kemudi pada mobil merupakan hal sangat penting karena berkaitan dengan keamanan dan keselamatan pengemudi. Pada sistem kemudi roda depan mobil punya masalah dalam pengendalian, hal ini disebabkan bobot mobil bertumpu di depan pada saat membelok. Akibatnya, ban depan mendapat beban lebih. Kondisi ini menyebabkan mobil dalam kondisi *understeer*.

Untuk itu dikembangkan sistem kemudi 4 roda yang dioperasikan secara manual maupun secara otomatis. Pengontrolan secara otomatis konvensional dengan memanfaatkan rangkaian pembanding mempunyai kelemahan yaitu memberi respons yang lambat, sehingga sistem kontrol kurang efektif bila mobil bergerak dengan kecepatan tinggi.

Digunakannya sistem pengaturan menggunakan Adaptif Fuzzy Logic Controller akan dapat memudahkan pengaturan sistem kemudi kendaraan. Keuntungan lain adalah mempunyai respons yang lebih cepat dan hasil yang presisi, dan mampu menyesuaikan diri terhadap lingkungan.

KATA PENGANTAR

Segala puji dan syukur penyusun panjatkan kehadirat Allah S.W.T. karena atas berkat rahmat dan hidayah-Nya kami dapat menyelesaikan Tugas Akhir ini dengan judul :

PENGONTROL SISTEM KEMUDI 4 RODA PADA MOBIL DENGAN MENGGUNAKAN ADAPTIF FUZZY LOGIC CONTROLLER

Tugas Akhir ini merupakan salah satu prasyarat untuk menyelesaikan studi di Bidang Studi Elektronika Jurusan Teknik Elektro, Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya dengan beban 6 SKS.

Dalam mengerjakan tugas akhir ini penulis mengacu pada teori yang didapat selama kuliah, literatur juga arahan dari dosen pembimbing telah banyak membantu.

Ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada :

- Ir. Nawantowibowo H selaku dosen Pembimbing
- Pujiono,ST selaku dosen Pembimbing
- Ir. Soetikno selaku koordinator Bidang Studi Elektronika
- Ir. Harmani S selaku dosen wali
- Bapak dosen yang telah memberikan ilmu selama penulis berkuliah
- Seluruh Staf di Jurusan Teknik Elektro
- Orang tua, kakak, dan Adik M. Oktavia Vidiyanti yang telah memberi dorongan moral
- Dyah,Khuz,Lalo,Pa'i,Gogot,Unggul,Aris,Beny serta seluruh warga Elka.

Penulis berharap tugas akhir ini dapat bermanfaat dan dikembangkan semaksimal mungkin.

Surabaya, Juli 1998

Penulis

DAFTAR ISI

JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAK	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan	2
1.3. Pembatasan permasalahan	2
1.4. Metodologi	3
1.5. Sistematika	3
1.6. Relevansi	4
BAB II TEORI PENUNJANG	5
2.1. Pendahuluan tentang kestabilan kendaraan	5
2.2. Standar sistem kemudi 4 roda (4WS)	7
2.2.1. Sistem kemudi standar Honda	7
2.2.2. Sistem kemudi standar Mazda	7

2.3. Gerakan kendaraan	9
2.3.1. Gerakan pada keadaan ideal (ackerman condition)	10
2.3.2. Gerakan pada keadaan nyata (actual condition)	10
2.3.3. Pengertian oversteer, understeer, dan neutralsteer	11
2.4. Gerak melingkar	14
2.5. Prinsip logika fuzzy	16
2.6. Teori logika fuzzy	17
2.7. Proses logika fuzzy	19
2.8. Kontrol dengan logika fuzzy	22
2.9. Kontroler proportional, Integral, dan derivative	25
 BAB III FUZZY LOGIC CONTROLLER NLX220	 27
3.1. Pendahuluan	27
3.2. Deskripsi pin NLX220	29
3.3. Arsitektur NLX220	31
3.3.1. Pengembangan sistem logika fuzzy	31
3.3.2. Fungsi keanggotaan (Membership function)	32
3.3.3. Variabel fuzzy	36
3.3.4. Aturan fuzzy (Fuzzy rule)	36
3.3.5. Floataing membership function	38
3.4. Operasi NLX220	41
3.4.1. Fuzzifier	41
3.4.2. Pembaharuan data output	42

3.4.3. Defuzzifier	43
3.5 Organisasi memori	45
3.6. Pewaktuan (timing)	49
3.6.1. Pewaktuan operasi	49
3.6.2. Pewaktuan kontroller	50
3.6.3. Pewaktuan output	51
3.6.4 Operasi prescale	51
3.6.5. Mode inaktif	52
 BAB IV PERENCANAAN PERANGKAT KERAS	 53
4.1. Perencanaan sistem	53
4.2. Perencanaan hardware	54
4.2.1. Rangkaian sensor kecepatan	54
4.2.2. Sensor posisi	56
4.2.3. Unit fuzzy NLX220 system	57
4.2.4. Rangkaian penggerak kemudi roda belakang	58
4.2.5. Rangkaian Display	60
 BAB V PERENCANAAN PERANGKAT LUNAK	 61
 BAB VI PENGUJIAN DAN PENGUKURAN	 69
6.1 Pengujian alat	69
6.2 Pengujian sistem secara lengkap	70

6.3 Faktor adaptif	74
 BAB VII PENUTUP	 76
7.1 Kesimpulan	76
7.2 Saran-saran	77
 DAFTAR PUSTAKA	 78
 LAMPIRAN	 79

DAFTAR GAMBAR

Gambar 2.1	Kondisi keadaan dalam kecepatan rendah	6
Gambar 2.2	Kondisi keadaan dalam kecepatan tinggi	6
Gambar 2.3	Gerakan kemudi standar Honda	8
Gambar 2.4	Gerakan kemudi standar Mazda	8
Gambar 2.5	Gerakan kendaraan	9
Gambar 2.6	Arah gaya yang bekerja pada kendaraan saat berbelok	11
Gambar 2.7	Kondisi kendaraan oversteer	14
Gambar 2.8	Kondisi kendaraan understeer	14
Gambar 2.9	Lintasan gerak melingkar	15
Gambar 2.10	Fungsi keanggotaan himpunan	18
Gambar 2.11	Proses Fuzzification	20
Gambar 2.12	Blok diagram sistem kontrol logika fuzzy	24
Gambar 3.1	NLX220P dengan 28 pin	29
Gambar 3.2	Diagram blok NLX220	32
Gambar 3.3	Fungsi keanggotaan temperatur	33
Gambar 3.4	Tipe fungsi keanggotaan	34
Gambar 3.5	Fungsi keanggotaan kecepatan	35
Gambar 3.6	Fungsi keanggotaan overlap	35
Gambar 3.7	Fuzzifikasi temperatur input	36
Gambar 3.8	Fungsi keanggotaan mengambang	38
Gambar 3.9	Defuzifikasi immediate	44

Gambar 3.10	Defuzifikasi accumulate	44
Gambar 3.11	Pewaktuan I/O	50
Gambar 4.1	Blok diagram sistem	53
Gambar 4.2	Rangkaian opto-isolator	56
Gambar 4.3	Rangkaian frekuensi to voltage converter	56
Gambar 4.4	Rangkaian sensor posisi	57
Gambar 4.5	Rangkaian pembatas tegangan	58
Gambar 4.6	Rangkaian resonator osilator	58
Gambar 4.7	Rangkaian driver motor DC	59
Gambar 5.1	Diagram input output sistem	62
Gambar 5.2	Membership function untuk Speed input	64
Gambar 5.3	Membership function untuk Front input	65
Gambar 5.4	Membership function untuk Control input	66
Gambar 5.5	Membership function untuk Timer	66
Gambar 5.6	Diagram alir proses kerja rule	68

DAFTAR TABEL

Tabel 3.1	Organisasi memori	45
Tabel 3.2	Organisasi byte genap (Command byte) dalam NLX220	47
Tabel 3.3	Organisasi byte ganjil (Select byte) dalam NLX220	48
Tabel 6.1	Perubahan tegangan terhadap frekuensi	71
Tabel 6.2	Perubahan sudut belok terhadap kecepatan	72
Tabel 6.3	Perubahan sudut belok terhadap posisi kemudi	73
Tabel 6.4	Perubahan sudut belok untuk input kecepatan rendah	73
Tabel 6.5	Perubahan sudut belok untuk input kecepatan tinggi	74

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi kontrol menggunakan logika fuzzy masih merupakan hal yang relatif masih baru di dunia industri, sedangkan pada dunia ilmu pengetahuan logika fuzzy ini telah dijabarkan sejak lama. Pada perkembangannya, logika fuzzy ini memiliki beberapa keuntungan yang tidak terdapat pada logika Boolean, sehingga memungkinkan untuk diterapkan pada banyak sistem.

Penerapan logika fuzzy dalam sistem kontrol memang masih dalam taraf pengembangan, mengingat teknologi analog dan digital masih menguasai hampir sebagian besar sistem kontrol di industri. Dengan berbagai penelitian mengenai logika fuzzy maka logika ini memiliki kemungkinan besar untuk menggantikan sistem kontrol konvensional.

Dasar-dasar logika fuzzy telah dipelajari oleh banyak ilmuwan sebelumnya sebagai jawaban atas kelemahan logika Boolean, namun baru pada pertengahan tahun 1960-an Profesor Lofti Zadeh menjabarkannya secara terperinci. Logika fuzzy ini ternyata mampu menjawab ketidakmampuan logika Boolean dalam mengenali perubahan parameter yang tidak jelas.

Logika fuzzy menggunakan suatu cara pembedaan kondisi satu dengan yang lainnya seperti cara berpikir manusia. Pada dasarnya manusia tidak berpikir dengan membedakan suatu hal secara pasti benar atau salah, namun menggunakan

tingkatan-tingkatan perubahan antara benar dan salah, sehingga suatu hal akan memiliki tingkat kebenaran maupun tingkat kesalahan tertentu.

Berdasarkan prinsip diatas maka logika fuzzy ini dapat memudahkan dalam perancangan suatu sistem kontrol, dimana terdapat tidak hanya dua kondisi benar atau salah, namun memiliki berbagai kondisi kebenaran sesuai dengan sistem yang dikontrol.

1.2 Tujuan

Tujuan dari tugas akhir ini adalah untuk mempelajari dan merancang suatu peralatan untuk mengontrol sistem kemudi 4 roda pada mobil sehingga kendaraan bisa bergerak lebih fleksibel dan stabil. Peralatan ini akan mengambil data sudut belok roda depan dan kecepatan kendaraan, selanjutnya kedua data tersebut diproses untuk menentukan apakah kestabilan kendaraan dalam kondisi aman atau kritis. Jika kestabilan kendaraan dalam kondisi kritis maka peralatan akan mengatur posisi kemiringan roda belakang untuk mengembalikan kestabilan kendaraan dalam kondisi aman.

1.3 Pembatasan Permasalahan

Pada tugas akhir ini permasalahan dibatasi pada perancangan dan pembuatan kontrol sistem kemudi 4 roda untuk menjaga kestabilan dan fleksibilitas kendaraan menggunakan teknologi logika fuzzy. Pada aplikasinya dipilih fuzzy microcontroller chip NLX 220 sebagai prosesor yang akan mengendalikan seluruh proses kontrol.

1.4 Metodologi

Untuk mencapai tujuan yang direncanakan, maka dalam pengerjaan tugas akhir ini dilakukan langkah-langkah sebagai berikut:

Studi literatur mengenai kestabilan dan fleksibilitas kendaraan, sistem kemudi 4 roda, serta penggunaan logika fuzzy dalam proses kontrol suatu sistem. Kemudian merancang peralatan mekanik yang merupakan simulasi dari mobil, selanjutnya melakukan penyesuaian parameter dan karakteristik sehingga bisa mewakili kondisi sebenarnya.

Pada tahapan selanjutnya dilakukan perencanaan rangkaian, setelah rangkaian dirakit, dilakukan pengujian tiap-tiap bagian dan selanjutnya dilakukan integrasi.

Pembuatan perangkat lunak merupakan langkah berikutnya yang berupa pembuatan rule-rule untuk proses fuzzy. Langkah selanjutnya adalah melakukan pengujian dan pengambilan data output.

Setelah selesai maka seluruh prosedur diatas disusun dalam suatu laporan tugas akhir.

1.5 Sistematika

Sistematika pembahasan dalam tugas akhir ini adalah sebagai berikut:

- Bab I pendahuluan yang berisi latar belakang, tujuan, pembatasan masalah, metodologi, sistematika dan relevansi dari tugas akhir.
- Bab II menjelaskan teori penunjang yang diperlukan dalam tugas akhir ini meliputi teori kestabilan kendaraan, sensor, dan teori fuzzy untuk kontrol.

Bab III menjelaskan teori khusus mengenai kontroler fuzzy yang digunakan.

Bab IV menjelaskan perencanaan perangkat keras meliputi perancangan sensor kecepatan dan sudut belok roda, sistem untuk NLX 220.

Bab V menjelaskan perencanaan dan pembuatan rule fuzzy untuk mendeteksi input dan menentukan aksi yang harus dilakukan.

Bab VI membahas mengenai hasil pengujian dan pengukuran

Bab VII merupakan bab penutup yang berupa kesimpulan dan saran dari tugas akhir.

1.6 Relevansi

Diharapkan dengan data dan hasil perencanaan tugas akhir ini dapat memberikan sumbangan tentang penerapan logika fuzzy dalam sistem pengendalian proses yang dapat dimanfaatkan secara luas dalam industri dan pengembangan ilmu khususnya mengenai teori fuzzy di teknik elektro ITS.

BAB II

TEORI PENUNJANG

2.1 Pendahuluan tentang kestabilan kendaraan

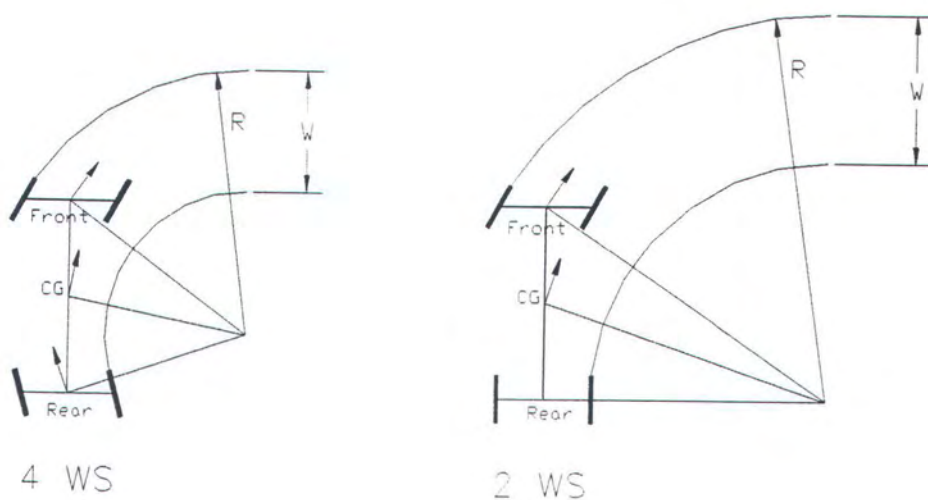
Faktor keamanan merupakan syarat utama dalam perancangan sebuah kendaraan bermotor, salah satu bagian yang menentukan keamanan adalah sistem kemudi kendaraan. Pada perancangan sistem kemudi ini yang perlu diperhatikan adalah tercapainya kestabilan kendaraan saat berbelok atau kondisi ideal (ackerman conditon), tetapi karena faktor kelembaman kendaraan dan kondisi medan jalan maka keadaan tersebut tidak tercapai.

Dengan mengamati kondisi tersebut maka dikembangkan sistem kemudi 4 roda (4WS), dimana posisi roda belakang juga dikendalikan untuk mencapai kestabilan kendaraan saat berbelok. Keuntungan lainnya dari sistem kemudi 4 roda (4WS) adalah kendaraan akan menggunakan ruang gerak yang relatif lebih sempit dibandingkan sistem kemudi konvensional.

Pada sistem kemudi 4 roda ini, posisi sudut belok roda belakang merupakan fungsi dari kecepatan kendaraan dan sudut belok roda depan. Dengan mengambil anggapan bahwa pada kecepatan rendah kendaraan akan mempunyai sudut belok yang tajam dan sebaliknya maka diambil kesimpulan bahwa kecepatan kendaraan berbanding terbalik dengan sudut belok kendaraan.

Pada kecepatan rendah sudut belok roda belakang berlawanan arah dengan sudut belok roda depan, sedangkan pada kecepatan tinggi sudut roda belakang bergerak searah dengan sudut belok roda depan.

Berikut ini adalah perbandingan sistem kemudi 2 roda dan sistem kemudi 4 roda.



Gambar 2.1 Kondisi kendaraan dalam kecepatan rendah



Gambar 2.2 Kondisi kendaraan dalam kecepatan tinggi

2.2 Standar sistem kemudi 4 roda (4WS)

Dalam perkembangannya perusahaan otomotif yang memproduksi kendaraan menentukan standar gerakan kemudi 4 roda, sistem standar yang berlaku saat ini ada 2 macam yaitu standar Honda dan standar Mazda.

2.2.1 Sistem kemudi standar Honda

Pada sistem kemudi 4 roda standar Honda gerakan roda belakang merupakan fungsi dari gerakan kemudi roda depan dengan mengikuti aturan sebagai berikut:

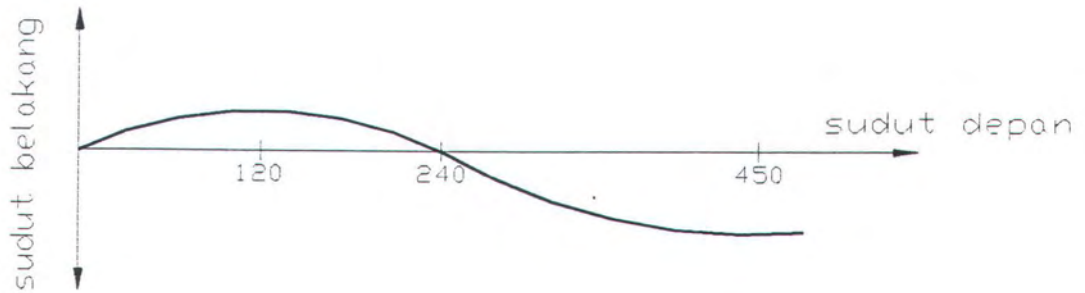
- Pada sudut putar kemudi 0° - 120° , kemudi roda belakang akan bergerak berlawanan dengan arah kemudi roda depan dengan perubahan sebesar 1.5° tiap 10° putaran kemudi roda depan.
- Pada sudut putar kemudi 120° - 240° , kemudi roda belakang akan bergerak menuju kearah netral dengan perubahan sebesar 1.5° tiap 10° putaran kemudi roda depan.
- Pada sudut putar kemudi 240° - 450° , kemudi roda belakang akan bergerak searah dengan arah kemudi roda depan.

Arah maksimum sudut belok roda depan 35° dan sudut belok roda belakang 5.3°

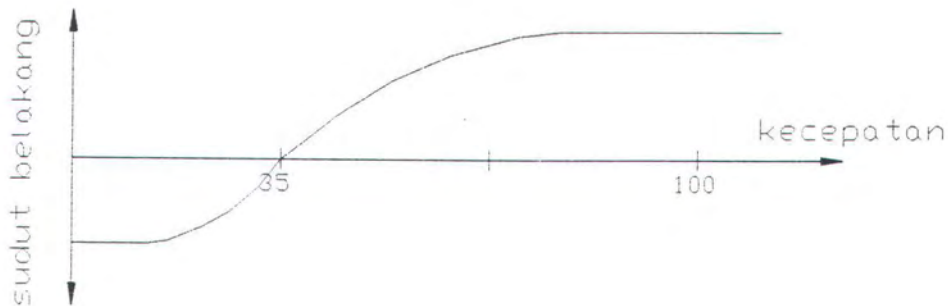
2.2.2 Sistem kemudi standar Mazda

Pada sistem kemudi 4 roda standar Mazda gerakan roda belakang merupakan fungsi dari laju kecepatan kendaraan dengan mengikuti aturan sebagai berikut:

- Pada kecepatan < 35 km/jam, kemudi roda belakang akan bergerak berlawanan dengan arah kemudi roda depan.
- Pada kecepatan 35 km/jam kemudi berada pada posisi normal 0.
- Pada kecepatan > 35 km/jam, kemudi roda belakang akan bergerak searah dengan arah kemudi roda depan.



Gambar 2.3 Gerakan kemudi standar Honda



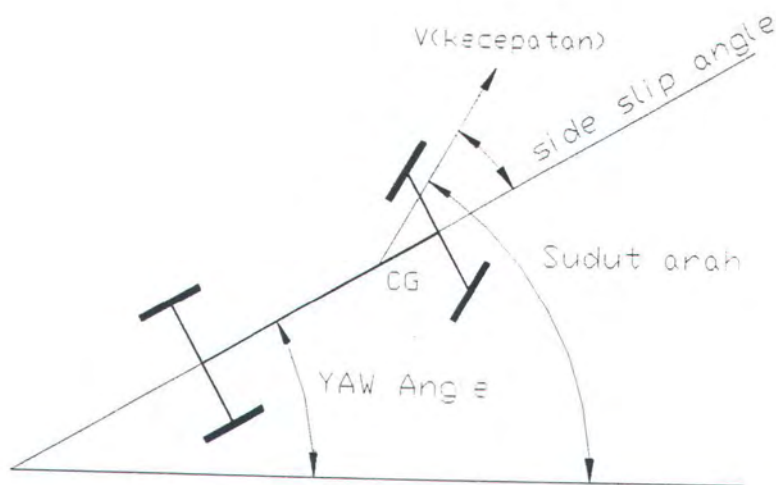
Gambar 2.4 Gerakan kemudi standar Mazda

2.3 Gerakan kendaraan

Untuk menentukan apakah posisi kendaraan dalam kondisi kestabilan aman atau kestabilan kritis perlu ditinjau beberapa parameter yang menjadi acuan pokoknya.

Parameter-parameter tersebut diantaranya adalah:

1. Simpangan (trayektori = Y) adalah arah simpangan posisi kendaraan setelah bergerak dalam suatu interval tertentu.
2. Sudut body (body angle = θ) adalah sudut antara sumbu longitudinal kendaraan dengan sumbu longitudinal jalan.
3. Sudut menyamping (side slip angle = β) adalah sudut arah gerakan nyata dari titik berat kendaraan dengan sumbu longitudinal kendaraan.



Gambar 2.5 Gerakan kendaraan

Jika sudut arah, sudut side slip dan sudut Yaw diturunkan terhadap waktu (δt) maka akan didapat kecepatan sudut yang masing-masing turn velocity (ωt), side slip velocity (ωs), dan Yaw velocity (ωy). Besarnya turn velocity sama dengan jumlah dari slip velocity dan Yaw velocity.

$$\omega t = \omega s + \omega y$$

Maka radius putar dari kendaraan saat belok adalah

$$R = \frac{V}{\omega \bullet t}$$

Dari beberapa parameter diatas kita dapat menentukan kondisi kestabilan kendaraan dengan menentukan sudut belok yang ideal.

2.3.1 Keadaan ideal (ackerman condition)

Adalah suatu kondisi kendaraan dimana $\omega s=0$ sehingga diperoleh harga turn velocity adalah

$$\omega t = \omega y$$

Jadi pada keadaan ideal kecepatan sudut belok dari kendaraan

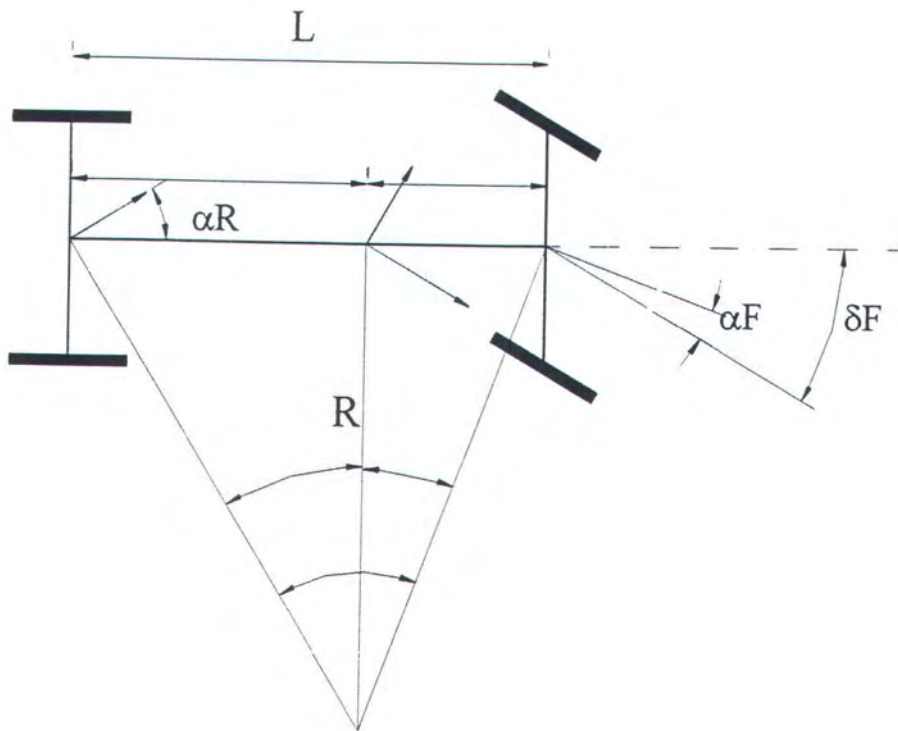
$$\omega y = \frac{V}{R_{ideal}}$$

2.3.2 Keadaan nyata (actual condition)

Untuk mendapatkan radius putar dari kendaraan haruslah memperhatikan faktor sudut side slip dan sudut slip dari masing-masing keadaan yang terjadi pada kendaraan pada saat belok sehingga radius putar nyata dari kendaraan bisa lebih

besar atau lebih kecil dari radius ideal, tergantung pada respon yang diberikan oleh kendaraan tersebut.

2.4 Pengertian oversteer, understeer, dan neuralsteer



Gambar 2.6 Arah gaya yang bekerja pada kendaraan saat berbelok

Gaya sentrifugal yang bekerja pada titik berat kendaraan akibat gerakan membelok dari kendaraan dengan kecepatan sedang atau tinggi memiliki pengaruh yang sangat besar terhadap gaya pada ban. Akibat gaya sentrifugal tersebut maka pada masing-masing ban akan membentuk gaya belok yang melawan besarnya gaya sentrifugal, pengaruhnya akan membentuk sudut slip (α) pada masing-masing ban.

Karakteristik pengendalian tergantung pada hubungan slip roda depan (α_f) dan roda belakang (α_r), sudut steer depan (δ_f), radius putar kendaraan (R), jarak poros depan dan belakang (L).

$$\delta_f - \alpha_f + \alpha_r = \frac{L}{R}$$

$$\delta_f = \frac{L}{R} + \alpha_f - \alpha_r$$

Gaya samping pada ban depan F_{yf} dan gaya samping pada ban belakang F_{yr} dapat dihitung dari kesetimbangan gaya kearah lateral pada kendaraan.

Pada kondisi ackerman dimana input sudut steer depan (δ_f) kecil sekali maka

$$F_{yf} = \frac{W_t \cdot V^2 \cdot L_2}{g \cdot R \cdot L}$$

$$F_{yr} = \frac{W_t \cdot V^2 \cdot L_1}{g \cdot R \cdot L}$$

W_t = Berat total kendaraan

g = Gravitasi

V = Laju kendaraan

L_1 = Jarak poros depan ke titik berat total kendaraan

L_2 = Jarak poros belakang ke titik berat total kendaraan

$$W_f = \frac{W_t \cdot L_2}{2 \cdot L}$$

$$W_r = \frac{W_t \cdot L_1}{2 \cdot L}$$

$$F_{yf} = \frac{2 \cdot W_f \cdot V^2}{g \cdot R}$$

$$F_{yr} = \frac{2 \cdot W_r \cdot V^2}{g \cdot R}$$

Sudut slip ban depan (α_f) dan ban belakang (α_r) dihitung dengan rumus sebagai berikut

$$\alpha_f = \frac{F_{yf}}{2 \cdot C_{\alpha f}} = \frac{W_f \cdot V^2}{C_{\alpha f} \cdot g \cdot R}$$

$$\alpha_r = \frac{F_{yr}}{2 \cdot C_{\alpha r}} = \frac{W_r \cdot V^2}{C_{\alpha r} \cdot g \cdot R}$$

$C_{\alpha f}$ dan $C_{\alpha r}$ adalah cornering stiffness dari ban depan dan ban belakang.

$$\delta_f = \frac{L}{R} + \frac{\frac{W_f}{\alpha_r} \cdot V^2}{g \cdot R}$$

$$\delta_f = \frac{L}{R} + \frac{K_{\mu s} \cdot V^2}{g \cdot R}$$

$K_{\mu s}$ = Koefisien understeer (rad)

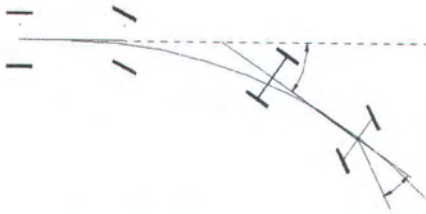
Jika kendaraan memiliki harga koefisien understeer $K_{\mu s}$ kurang dari 0 berarti sudut slip ban depan kurang dari sudut slip ban belakang kondisi seperti ini dinamakan over steer. Jika kendaraan memiliki harga koefisien understeer $K_{\mu s}$ lebih dari 0 berarti sudut slip ban depan lebih dari sudut slip ban belakang kondisi seperti ini dinamakan under steer.

Gejala neutral steer dicapai jika harga koefisien under steer $K_{\mu s}$ sama dengan 0, berarti sudut slip ban depan sama dengan sudut slip ban belakang. Dalam

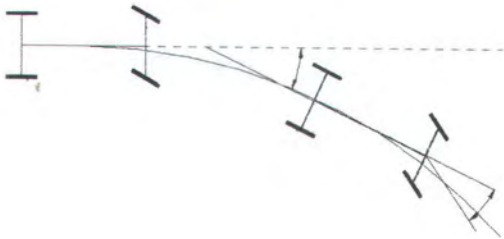


hal ini sudut slip ban depan hanya merupakan fungsi dari radius putar dan jarak poros roda saja.

$$\delta f = \frac{L}{R}$$



Gambar 2.7 Kondisi kendaraan oversteer

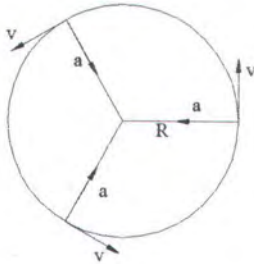


Gambar 2.8 Kondisi kendaraan understeer

2.4 Gerak melingkar

Untuk sebuah benda yang bergerak melingkar dengan laju konstan, arah vektor kecepatan berubah terus menerus, tetapi besarnya tidak. Pada Gambar 2.9 memperlihatkan hubungan sesaat antara a dan v di beberapa titik dalam gerakan. Besar dari v tetap tetapi arahnya senantiasa berubah. Hal ini memberikan percepatan yang besarnya juga tetap dan arahnya terus menerus berubah. Kecepatan v selalu menyinggung lingkaran dalam arah tegak, sedangkan

percepatan a selalu radial ke dalam. Itu sebabnya percepatan a disebut percepatan radial atau percepatan sentripetal.



Gambar 2.9 lintasan gerak melingkar

Besar percepatan sentripetal adalah perubahan besar kecepatan tiap satuan waktu.

$$a = \frac{V^2}{R}$$

Bila benda tersebut adalah sebuah mobil yang mempunyai massa m maka pada mobil tersebut akan bekerja gaya sentripetal sebesar

$$F = m * \frac{V^2}{R}$$

Gaya sentripetal ini ditimbulkan oleh gaya gesekan yang berarah ke samping, yang ditimbulkan oleh jalan pada roda. Besarnya gaya gesekan tersebut adalah

$$f_k = \mu_k \bullet mg$$

Dengan mensubstitusi kedua persamaan gaya sentripetal tersebut maka diperoleh

$$m * \frac{V^2}{R} = \mu_k * mg \quad \text{sehingga} \quad \mu_k = \frac{V^2}{gR}$$

Untuk memperbesar gaya sentripetal yang bekerja pada mobil maka koefisien gesek antar mobil dengan jalan diperbesar. Hal ini dilakukan dengan mengatur posisi roda terhadap jalan.

2.5 Prinsip logika fuzzy

Pada tahun 1965 Profesor Lofti Zadeh dari University of California menyatakan bahwa suatu hal tidak dapat dinyatakan dengan pasti namun dapat memiliki kondisi diantara keduanya. Kondisi *true* atau *false* tidak dapat menerangkan banyak kejadian di alam semesta, dimana kebanyakan kejadian tersebut merupakan dari benar ke salah (atau sebaliknya) dan bukan hanya bernilai benar atau salah.

Untuk menerangkan perubahan yang tidak tentu dari benara ke salah tersebut, Zadeh mengemukakan teori yang memperluas himpunan klasik menjadi apa yang dinamakan himpunan fuzzy. Logika fuzzy merupakan logika dengan nilai banyak (multi valued logic). Sehingga logika fuzzy mengijinkan adanya tingkat keanggotaan (degree of membership) dari suatu anggota himpunan bernilai antara 0 dan 1 , sehingga sesuatu dapat bernilai sebagian benar dan sebagian salah dalam satu waktu. Bart Kosko telah membuktikan bahwa logika Boolean adalah satu kasus khusus dari logika fuzzy.

Contoh sederhana tentang prinsip fuzzy adalah pada pembagian kecepatan kendaraan, misalnya sebagai berikut

kecepatan rendah	$< 35 \text{ km/jam}$
kecepatan menengah	35 km/jam sampai 55 km/jam
kecepatan tinggi	$\geq 56 \text{ km/jam}$

Jika menggunakan prinsip Boole maka terlihat bahwa kecepatan 55 dan 56 memiliki klasifikasi yang berbeda, padahal kedua kecepatan tersebut tidak memiliki beda yang banyak. Hal ini memperlihatkan kelemahan logika Boole untuk

mengklasifikasikan suatu anggota yang terletak pada titik perbatasan, lagipula penentuan batas tersebut banyak bergantung pada tiap individu, karena seseorang mungkin saja memiliki klasifikasi dan nilai batas yang berbeda. Pada fuzzy kecepatan 55 dan 56 tidak dibedakan dengan jelas namun samar, sehingga kecepatan 55 dapat dianggap terletak antar kecepatan menengah dan kecepatan tinggi begitu pula dengan kecepatan 56. Jadi kecepatan 55 dan 56 memiliki tingkat keanggotaan yang berbeda, beda tingkat keanggotaan tersebut akan menentukan kecenderungan klasifikasinya. Karena itu kecepatan 55 dapat dikatakan lebih cenderung sebagai kecepatan menengah sedangkan kecepatan 56 cenderung kecepatan tinggi. Dengan makin bertambah atau berkurangnya kecepatan maka tingkat keanggotaannya untuk suatu himpunan akan makin bertambah sehingga lebih jelas kondisinya.

2.6 Teori Logika Fuzzy

Pengembangan logika fuzzy dilakukan untuk menjawab permasalahan two valued logic yaitu logika yang hanya mengenal suatu keadaan dalam 2 kemungkinan, seperti benar atau salah. Teori two valued logic terbukti efektif untuk memecahkan masalah dengan syarat permasalahannya dapat dideskripsikan kuantitasnya secara tepat.

Sebagai contoh adalah menentukan apakah orang dengan tinggi 160 cm termasuk tinggi, sedang atau pendek. Apabila menggunakan metode two valued logic maka pemecahannya adalah memberi range dibawah 155 cm sebagai pendek dan diatas 155 cm sebagai tinggi, kesulitan yang pertama adalah menentukan batas

pendek dan tinggi secara kuantitatif, karena setiap orang mempunyai nilai yang berbeda. Kesulitan yang kedua adalah menentukan akan dimasukkan kemanakah nilai 155 cm tersebut (karena tepat berada pada perbatasan). Logika fuzzy membagi suatu keadaan dalam interval $[0,1]$ yang secara intuitif dapat dinyatakan dalam contoh diatas dengan mengelompokkan orang sebagai pendek, agak pendek, sedang, agak tinggi, dan tinggi. Jika logika fuzzy ini diterapkan pada permasalahan tinggi badan diatas maka dapat dikatakan bahwa rang dengan tinggi badan 155 cm mempunyai nilai kebenaran 0.7 pendek dan 0.3 tinggi. Hal ini akan memberikan kesimpulan bahwa orang tersebut cenderung pendek. Untuk lebih jelas, permasalahan tersebut dapat dilihat pada grafik berikut :



Gambar 2.10 Fungsi keanggotaan himpunan

Dalam logika fuzzy terdapat istilah-istilah sebagai berikut :

1. Fungsi keanggotaan (*membership function*) adalah fungsi yang memetakan masukan nyata (*crisp input*) dengan domainnya pada derajat keanggotaan.
2. Skala keanggotaan / Derajat keanggotaan (*degree of membership*) merupakan skala dimana nilai nyata masukan setara dengan fungsi keanggotaannya. Skala ini bernilai antara 0 sampai 1.

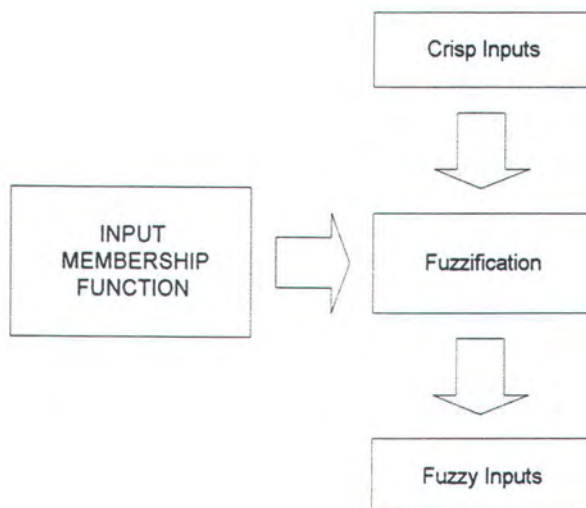
3. Nilai masukan (*crisp input*) adalah nilai masukan yang bernilai skalar, tertentu dan tunggal. Untuk sistem kontrol nilai ini dapat berupa masukan dari transducer. Misalnya kecepatan kendaraan 100 km/jam.
4. Label adalah nama yang merupakan penggambaran dari fungsi keanggotaan.
5. Domain adalah lebar total dari fungsi keanggotaan yang merupakan jangkauan dari suatu konsep angka-angka tertentu, dimana fungsi keanggotaan itu dipetakan.
6. Nilai tengah merupakan suatu nilai dari fungsi keanggotaan dimana mempunyai nilai kebenaran sempurna (1 atau 0).

2.7 Proses Logika Fuzzy

Dalam pemecahan masalah menggunakan logika fuzzy diperlukan tiga tahap proses yaitu:

- a. proses fuzzification yaitu mengubah variabel input yang berupa variabel crisp (besaran nyata berupa variabel yang berorientasi numerik) ke dalam variabel fuzzy.
- b. proses evaluasi aturan (*rule evaluation*) yaitu mencari nilai aksi (*action*) dengan memberikan bobot pada setiap aturan yang diberikan.
- c. proses defuzzification yaitu mengubah variabel fuzzy yang terbentuk dari proses evaluasi aturan menjadi variabel crisp.

Proses fuzzification adalah rangkaian usaha untuk mentransformasi variabel input yang mulanya bersifat numerik (crisp inputs) ke variabel fuzzy (fuzzy inputs). Transformasi ini dipengaruhi oleh fungsi keanggotaan yang digunakan.



Gambar 2.11 Proses Fuzzification

Proses evaluasi aturan (rule evaluation) adalah rangkaian usaha untuk menentukan nilai aksi sebagai tanggapan atas setiap input atau kombinasi input yang diberikan dengan memberi bobot pada masing-masing aturan yang telah ditetapkan. Di dalam proses ini terdapat dua komponen utama yaitu himpunan aturan (rule sets) dan metode evaluasi aturan.

Himpunan aturan (rule sets) adalah semua aturan yang diperlukan untuk menentukan tanggapan terhadap input atau kombinasi input yang diberikan. Aturan ini bersifat linguistik dan mempunyai bentuk “jika ... maka” (If then).

Bentuk aturan ini diciptakan berdasarkan keinginan untuk :

1. menyediakan cara yang mudah bagi para ahli untuk mengekspresikan pengetahuan dan pengalaman mereka.
2. menyediakan cara yang mudah bagi para disainer untuk menyusun dan memprogram aturan fuzzy.
3. mengurangi biaya disain.

Bentuk umum dari aturan fuzzy adalah : jika x_1 adalah A_{k1} dan x_2 adalah A_{k2} atau x_3 adalah A_{k3} ...maka y_1 adalah B_{k1} ; dimana x_1 , x_2 , dan x_3 adalah input kejadian 1 (antecedent1), kejadian 2, dan kejadian 3; A_{k1} , A_{k2} , dan A_{k3} adalah himpunan fuzzy yang berkorelasi dengan kejadian; y_1 adalah output kejadian dan B_{k1} adalah himpunan fuzzy yang berkorelasi dengan output.

Metode evaluasi aturan adalah metode yang digunakan dalam mengevaluasi aturan yang telah ditetapkan. Ada beberapa metode evaluasi aturan yang sering dipakai seperti mini rule (Mamdani), product rule (Larsen), Max-Min rule (Zadeh), Arithmetic rule (Zadeh) dan Boolean. Metode yang digunakan dalam tugas akhir ini adalah metode Max-Min sesuai dengan metode yang dipakai oleh Fuzzy Microcontroller NLX 220 yang digunakan. Konsep dari metode ini adalah mencari nilai minimum pada setiap rule, kemudian mencari nilai maksimum dari himpunan aturan yang berkorelasi dengan satu kejadian output sehingga dapat ditentukan nilai aksi yang seharusnya, demikian diulangi untuk setiap kejadian output. Nilai minimum pada setiap rule menggambarkan derajat ke-fuzzian aturan tersebut,

sedangkan nilai maksimum dari nilai minimum himpunan aturan yang berkorelasi dengan satu kejadian output menggambarkan kejadian yang paling dapat “dipercaya” karena mempunyai derajat ke-fuzzian paling tinggi sehingga aturan yang mempunyai derajat ke-fuzzian paling tinggi diambil sebagai aturan yang paling dapat dipercaya (the winning rule).

Proses defuzzification adalah tahap terakhir proses logika fuzzy yang berfungsi untuk mengubah output yang berupa output fuzzy menjadi output crisp. Ada beberapa metode defuzzification, yaitu : Center of Gravity (COG), fuzzy singleton, accumulate, dan immediate. Dalam tugas akhir ini metode yang digunakan adalah metode accumulate dan immediate sesuai dengan spesifikasi NLX220 yang digunakan.

Metode accumulate pada output berarti nilai output sama dengan nilai aksi aturan yang menang ditambah dengan nilai output sebelumnya sehingga metode ini dapat digunakan sebagai pendekatan proses integrasi. Sedangkan metode immediate berarti nilai output sama dengan nilai aksi aturan yang menang.

2.8 Kontrol Dengan Logika Fuzzy

Tujuan utama suatu sistem kontrol adalah menghasilkan suatu keluaran yang dikehendaki untuk setiap masukan yang diberikan. Rangkaian usaha yang dilakukan untuk mengolah masukan menjadi keluaran yang dikehendaki disebut dengan proses kontrol. Secara konvensional dikenal beberapa proses kontrol yaitu metoda look-up table, metode pemodelan secara matematis untuk mencari fungsi

transfer antara masukan dan keluaran, dan metode dengan menggunakan logika fuzzy. Metode look-up table efektif hanya jika sistem yang dikontrol hanya mempunyai resolusi yang rendah dan variabel masukan sedikit. Kelemahan utama sistem ini adalah jika sistem rumit maka akan menghabiskan banyak tempat di memori, menimbulkan gangguan pada sistem karena adanya event yang tidak tertanggapi, dan meloncatnya nilai keluaran dari tabel yang satu ke tabel yang lainnya.

Model matematis dari suatu sistem harus secara tepat menggambarkan perilaku sistem terhadap masukan yang diberikan. Untuk sistem yang rumit hal ini menjadikan pembuatan model matematis sangat sulit, dengan hasil suatu persamaan yang kompleks dan sulit diaplikasikan. Pembuatan model matematis juga membutuhkan keahlian khusus sehingga meningkatkan biaya desain. Kelemahan lain dari model ini adalah keterbatasan data yang ada sehingga tidak memungkinkan pembuatan model dan adanya asumsi-asumsi yang memudahkan pembuatan model matematis tapi sangat mengurangi keakuratan sistem.

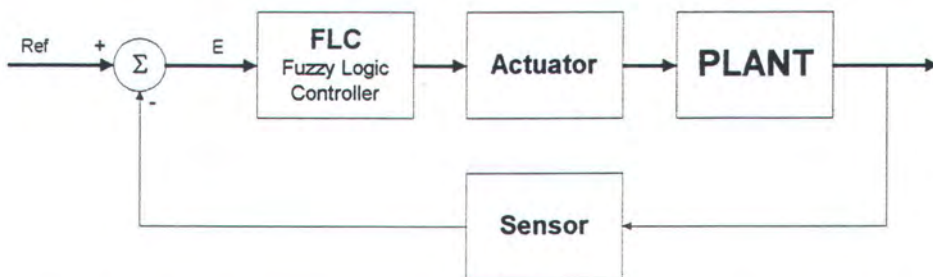
Metoda kontrol berdasarkan logika fuzzy dikembangkan untuk mengetasi kelemahan-kelemahan tersebut. Logika fuzzy menawarkan pemecahan masalah yang intuitif, dan disesuaikan dengan cara berfikir manusia. Penggunaan teknologi fuzzy dalam rekayasa proses dan sistem informasi akan menghasilkan alat-alat yang handal, tahan, luwes, dan lebih canggih dibandingkan dengan alat-alat digital biasa. Hal ini akan memproduksi sistem pengambil keputusan, sistem kontrol otomatis

yang akan membawa kepada mesin yang mempunyai daya pikir (*intelligent machine*).

Keuntungan lebih lanjut dari penerapan logika fuzzy dalam kontrol akan mempunyai keuntungan-keuntungan sebagai berikut :

- Kemudahan bagi pemakai yang lebih baik
- Kemampuan menyesuaikan diri yang lebih baik
- Kemampuan untuk diagnosa sendiri
- Kinerja yang lebih baik dengan konsumsi daya yang lebih rendah

Struktur dasar penggunaan logika fuzzy untuk suatu sistem kontrol, ditunjukkan seperti seperti gambar di bawah ini



Gambar 2.12 Blok Diagram Sistem Kontrol Logika Fuzzy

Sistem kontrol otomatis pada umumnya terdiri dari empat bagian utama : sensor, kontroler, aktuator, dan plant. Sensor berfungsi sebagai pengambil data perilaku dari sistem. Aktuator memberikan daya untuk menggerakkan peralatan yang dikontrol(plant) agar mencapai suatu harga yang diinginkan.

Kontroler berfungsi memberikan perintah ke aktuator sesuai aksi kontrol menurut besarnya deviasi (error) yaitu selisih antara referensi dan output yang terukur oleh sensor. Sistem kontrol yang digunakan adalah sistem kontrol berumpan balik (lup tertutup) karena sistem ini cenderung menjaga hubungan yang telah ditentukan antara keluaran dan masukan acuan dengan membandingkannya dan menggunakan selisihnya sebagai alat pengontrolan, secara umum sistem kontrol berumpan balik mempunyai keunggulan dibanding sistem kontrol tanpa umpan balik (lup terbuka).

2.9 Kontroler Proportional, Integral, dan Derivative

Salah satu metode kontrol linier yang populer adalah Proportional, Integral, dan Derivative atau PID. Kekurangan metode kontrol konvensional diperbaiki dengan kontrol fuzzy. Logika fuzzy lebih fleksibel dibanding metode kontrol linier dan dapat mencapai unjukkerja yang lebih baik khususnya pada sistem non-linier atau multi input maupun output dengan respon yang cepat.

Disini akan dijelaskan tentang pendekatan kontroler PID dengan menggunakan Kontroler Logika Fuzzy (KLF) dan menunjukkan bagaimana memodifikasinya sehingga dapat memperbaiki unjuk kerja dalam aplikasi-aplikasi khusus. Model ini merupakan merupakan awal bagi engineer kontrol agar lebih mengenal tentang kontrol PID menggunakan KLF. Output kontroler PID konvensional terdiri dari penjumlahan tiga komponen, dan masing-masing dikalikan dengan nilai koefisiennya.

Komponen pendukung tersebut adalah Proportional, Integral, dan Derivative. Model logika fuzzy dapat dibuat mirip dengan proses ini dan memiliki keuntungan tambahan. Secara matematis, bentuk proporsional mempunyai fungsi alih berupa hasil perkalian antara error dengan suatu konstanta, sedangkan integral merupakan penjumlahan secara terus-menerus dari error. Bentuk derivative merupakan selisih error sekarang dengan sinyal error sebelumnya. Error disini berupa suatu nilai selisih antara nilai yang diinginkan dan nilai sebenarnya, nilai ini dapat bertambah atau berkurang di sekitar titik yang diinginkan.

Pada fuzzy hal ini dapat didekati dengan cara sebagai berikut:

- Proporsional, didekati dengan memberikan setiap konstanta aksi pada masing-masing variabel fuzzy yang berhubungan dengan error.
- Integral, didekati dengan menggunakan metode accumulate, yaitu dengan menjumlahkan setiap konstanta aksi sebelumnya pada variabel fuzzy yang berhubungan dengan error .
- Derivative, didekati dengan memberika konstanta aksi pada variabel fuzzy yang berhubungan dengan selisih error sekarang dengan sebelumnya.

Ketiga unsur tersebut kemudian dijumlahkan sehingga diperoleh pendekatan kontrol PID. Keuntungan pemakaian fuzzy pada sistem ini adalah adanya kemudahan desain dan penentuan nilai aksi yang hanya berlaku lokal, jadi jika ada satu bagian yang mengalami perubahan maka bagian yang lain tidak terpengaruh.

BAB III

FUZZY LOGIC CONTROLLER NLX220¹

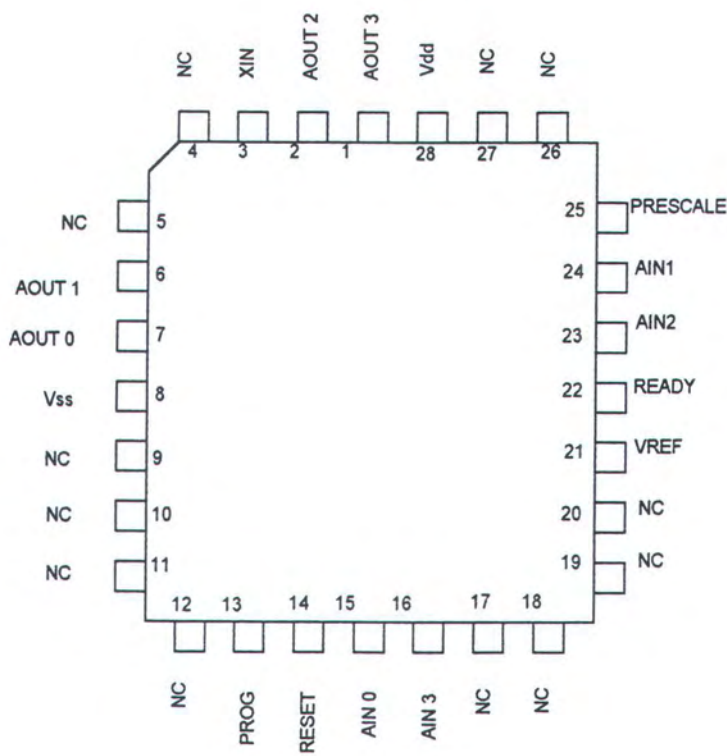
3.1 Pendahuluan

Merupakan Kontroler Logika Fuzzy (KLF) stand-alone, single chip, fleksibel, mudah beradaptasi, bisa bekerja dengan eksternal ROM (EEPROM) atau One time Program (OTP), mempunyai empat input analog 8-bit, empat output analog 8-bit, 28 pin, dapat menggunakan enam tipe membership function, 111 variable fuzzy, dan 50 rule. Dalam aplikasinya dapat untuk memenejemen keperluan daya, pengendalian motor, pengendalian pemanas, pengendalian mobil dan kontrol proses industri.

NLX220 mempunyai unjuk kerja tinggi dan merupakan kontroler logika fuzzy stand alone. Perangkat tersebut dapat menghitung logika fuzzy secara langsung dalam hardware. NLX220 adalah kontroler yang memiliki keunggulan dalam hal mudah pemakaian, unjuk kerja, terencana dan robust atau kuat terhadap kondisi lingkungan operasi yang kasar. Perangkat tersebut berisi empat input analog 8-bit dan generator clock ada didalam atau internal. NLX220 memakai daya yang kecil saat operasi normal dan mempunyai mode down power sehingga mereduksi daya dengan faktor 10. Logika fuzzy adalah memproses daya secara metodologi sehingga dengan mudah mengakomodasi data input dan input-input sistem

sehingga dengan mudah mengakomodasi data input dan input-input sistem nonlinier untuk pengembangan yang cepat sistem pengendalian robust. Metodologi menggunakan penjelasan bahasa dari sistem, sehingga hal tersebut menjadi intuitif dan sederhana dalam pemakaiannya. Logika fuzzy dapat digunakan secara tidak mahal dan dapat menambah kemampuan berpikir pada berbagai produk. Logika fuzzy dapat memperbaiki unjuk kerja, penambahan perencanaan dan menaikkan efisiensi. Memori penyimpanan fungsi keanggotaan fuzzy dan parameter-parameter rule. Organisasi memori adalah fleksibel dan secara efisien mengadaptasi terhadap berbagai keperluan aplikasi. Perangkat tersebut dapat menyimpan 111 variabel fuzzy, yang diorganisasikan menjadi rule seperti yang dikehendaki. Perangkat tersebut memiliki enam tipe yang berbeda fungsi keanggotaan untuk keperluan aplikasi. Fungsi keanggotaan adalah berupa slop yang konstan dan hanya perlu spesifikasi tipe, lebar atau width dan center. NLX220 memiliki fungsi keanggotaan mengambang. Center dan width dari fungsi keanggotaan dapat dibuat “mengapung” atau bervariasi secara dinamis. Fungsi keanggotaan mengapung dapat digunakan untuk mengukur derivative, membangun timer atau untuk mendrift sensor. Ada dua metode defuzzifikasi yaitu immediate dan accumulate. Mode immediate mendrive sebuah output dengan nilai yang spesifik. Mode accumulate menambahkan pada nilai output sebelumnya. Informasi aplikasi dimasukkan secara mudah menggunakan sistem pengembangan

INSIGHT™ yang dapat dijalankan pada Window. Diperlukan sedikit pengetahuan tentang logika fuzzy untuk dapat menggunakan perangkat atau sistem tersebut.



Gambar 3.1 NLX220P dengan 28 pin.

3.2 Deskripsi pin NLX220

Input

Reset

Sebuah signal aktif low yang menginisialisasi perangkat. RESET harus tetap aktif untuk paling sedikit delapan siklus clock untuk menjamin operasi yang benar.

RESET dapat didrive power up delayed circuit. Dengan penjelasan RESET ini maka NLX220 seluruhnya memakai mode power rendah.

AIN(3:0)(Analog Input Data).

Data analog dirubah secara internal menjadi data digital 8-bit. Input-input yang tak terpakai harus dihubungkan ke ground.

Xin (Clock Input)

Dapat didrive oleh clock external atau oleh kristal sedang lead yang lain dihubungkan keground.

Prog

Pin ini digunakan untuk pemrograman NLX220P. Pin ini tidak ada pada NLX 220 maka pin ini dalam operasinya harus dihubungkan ke ground.

Prescale

Level logika satu menyebabkan perangkat ke mode prescale sedang zero menyebabkan operasi normal.. Pin ini dapat diground jika mode prescale tidak dipakai atau dapat juga dihubungkan ke pin READY untuk pemakaian seterusnya. Mode juga dapat diminta selama operasi melalui logika external. Setelah RESET tidak dipertahankan, pin prescale harus dipertahankan pada logika rendah untuk selama paling sedikit empat siklus clock.

Output

Aout(3:) Analog Output Data.

Data digital delapan bit dirubah secara internal menjadi sebuah level analog.

Ready

Setelah reset, signal pin ini yang mana perangkat tersebut adalah start untuk mensampel dan memproses data. Pin harus tak dihubungkan disebelah kiri atau diikat menjadi PRESCALE selama operasi.

Vref

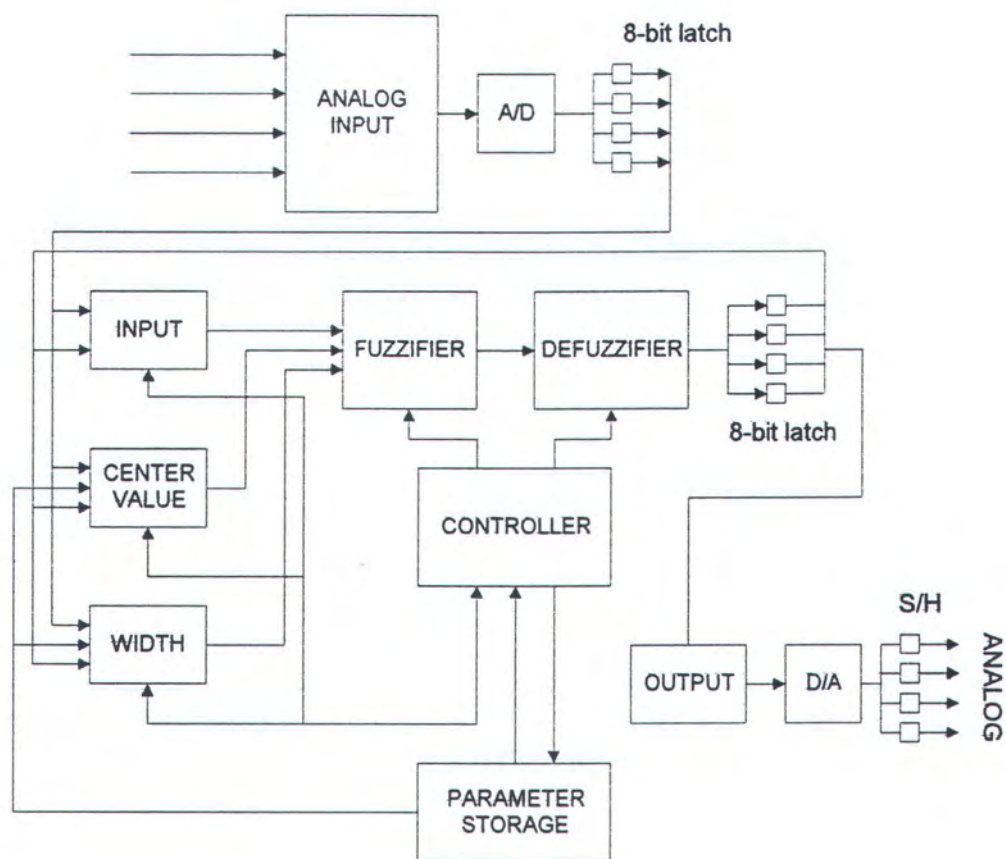
Filter tegangan reference internal , harus dihubungkan keground melalui kapasitor 0,1 μ F.

3.3 Arsitektur NLX220

Diagram NLX220 seperti ditunjukkan pada gambar 3.2 dibawah. Elemen-elemen utama adalah Fuzzifier, Defuzzifier dan Kontroler. Fuzzifier merubah data input menjadi data fuzzy. Fuzzifier yang berhubungan dengan kontroler adalah mengevaluasi data fuzzy melalui himpunan rule yang didefinisikan oleh pemakai untuk menjelaskan bagaimana sistem bekerja. Ketika rule-rule telah dievaluasi, Defuzzifier menunjukkan suatu nilai aksi output yang dikehendaki.

Untuk mengerti operasi perangkat, yang perlu dipahami adalah bagaimana memasukkan model logika fuzzy dan bagaimana melakukan kalkulasi logika fuzzy. Setelah proses kalkulasi selesai, tahapan selanjutnya adalah menentukan rule mana yang menang. Rule yang menang akan menentukan aksi yang akan dilakukan oleh controller.

Seksi berikut adalah menjelaskan konsep dasar sistem fuzzy NLX220.



Gambar 3.2 Diagram Blok NLX220

3.3.2 Fungsi Keanggotaan (Membership Function)

Fungsi keanggotaan digunakan untuk membagi-bagi menjadi seksi-seksi seluruh range yang mana sebuah input dapat bervariasi menjadi range yang mana. Fungsi keanggotaan dibandingkan dengan data input untuk melihat termasuk yang manakah data tersebut. Mereka mempunyai nama yang dipilih oleh perancangnya sendiri, seperti panas, cepat atau tinggi, untuk mengklasifikasikan data tersebut.

Thermometer dapat digunakan untuk menggambarkan konsep fungsi keanggotaan dan menunjukkan bagaimana logika fuzzy bekerja seperti manusia.

Dengan membagi-bagi range thermometer menurut kenyamanan sebagai berikut:

Dibawah 60°F = Cold (dingin)

60°F sampai 70°F = Cool (sejuk)

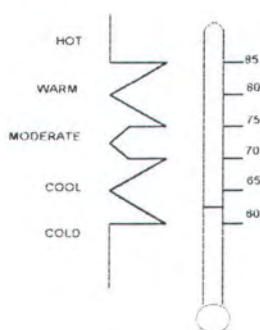
70°F sampai 75°F = Moderate (sedang)

75°F sampai 85°F = Warm (hangat)

Diatas 85°F = Hot (panas)

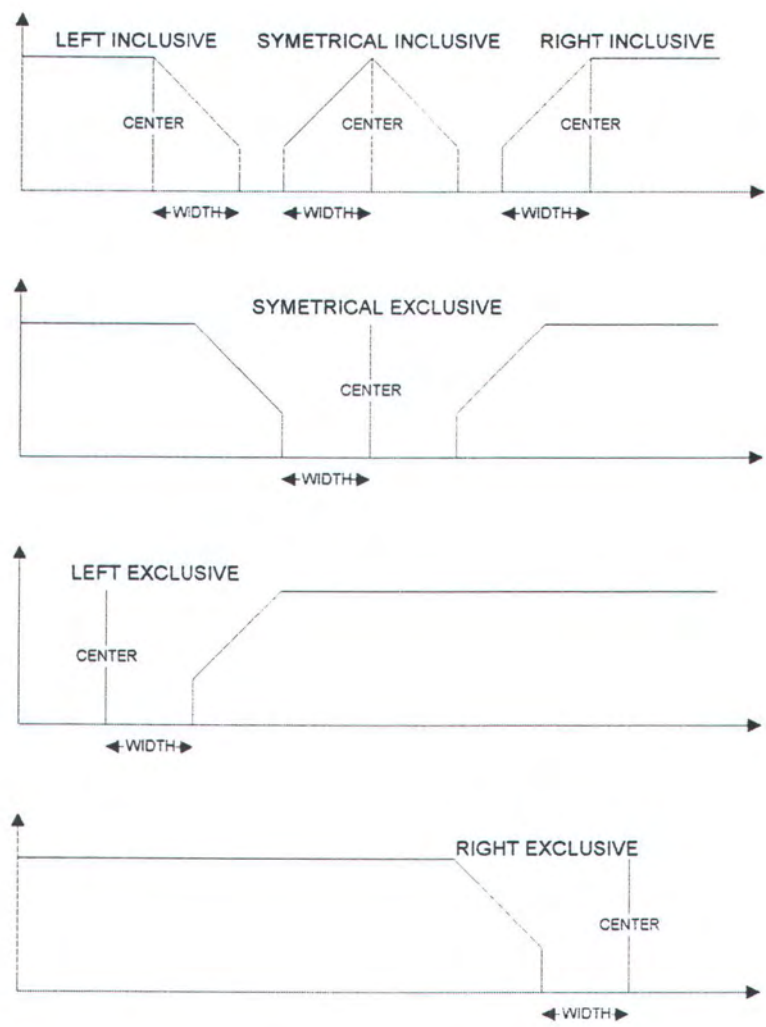
Pembagian tersebut adalah suatu cara intuisi seseorang untuk membag-bagi temperatur karena mereka berdasarkan perasaan. Seseorang dapat menjelaskan bahwa ruang dengan suhu 60°F adalah sejuk. Dalam logika fuzzy lima pembagian tersebut disebut fungsi keanggotaan dan digambarkan pada gambar 3.4. Fungsi keanggotaan dapat pisah-pisahkan seperti ditunjukkan atau mereka bisa saling tindih atau overlap. Hal tersebut memungkinkan data jatuh didaerah overlap yang termasuk di dua fungsi keanggotaan.

Suhu sebagai contoh dapat diterangkan agak sejuk dan agak dingin.



Gambar 3.3 Fungsi Keanggotaan Temperatur

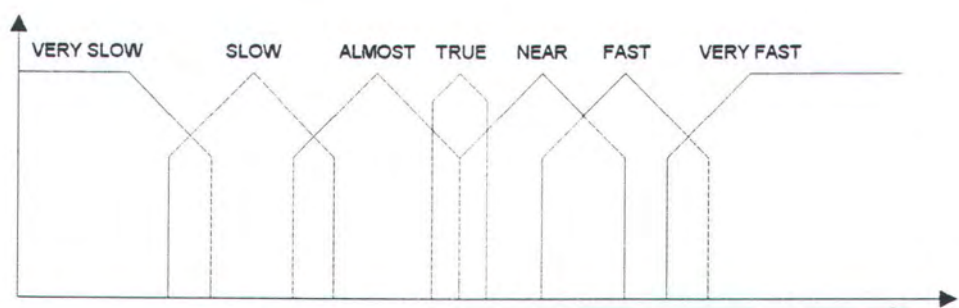
NLX220 memiliki enam fungsi keanggotaan yang berbeda dan slope yang konstan seperti gambar 3.4. Hal tersebut terdiri dari fungsi Left, Symetrical dan Right Inclusive dan sebaliknya dalam Fungsi Exclusive.



Gambar 3.4 Tipe Fungsi Keanggotaan

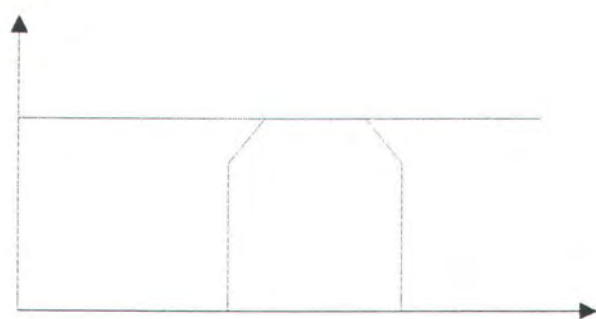
Dalam aplikasinya hal tersebut perlu diberi nama, tipe spesifikasi bentuk dan juga nilai angka untuk center dan width.

Untuk pengendalian yang teliti dapat dibuat fungsi keanggotaan inclusive simetrik secara sempit. Contoh aplikasi pengendalian motor yang memerlukan ketelitian. Pada aplikasi tersebut digunakan berbagai type fungsi keanggotaan yang berbeda lebarnya.



Gambar 3.5 Fungsi Keanggotaan Kecepatan.

Fungsi keanggotaan tersebut dapat overlap untuk menghasilkan bentuk baru seperti trapesium seperti gambar 3.6. Trapesium tersebut dibentuk oleh adanya fungsi keanggotaan Left dan right inclusive yang overlap. Input dapat masuk ke trapesium yang merupakan dua fungsi keanggotaan.



Gambar 3.6 Fungsi Keanggotaan Overlap.

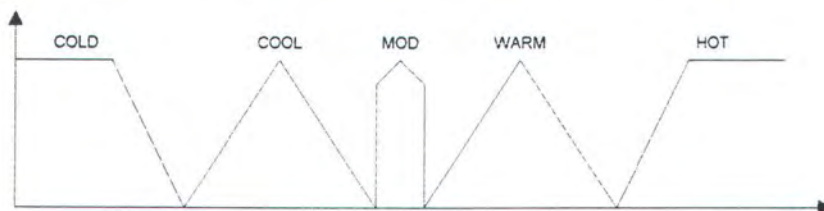
3.3.3 Variable Fuzzy

Variable fuzzy adalah pernyataan bahasa yang menggambarkan hubungan antara nilai input terhadap fungsi keanggotaan yang meliputi seluruh sumbu. Variabel fuzzy mereferensi sebuah fungsi keanggotaan dan sebuah nilai variabel input. Sebagai contoh sebuah variabel fuzzy sebagai berikut:

If Temperatur is Cool

Dalam contoh ini, 'Temperature' menunjukkan sebuah input dan 'Cool' adalah sebuah fungsi keanggotaan.

Asosiasi ini dilakukan oleh Fuzzifier. Hasilnya adalah sebuah nilai data fuzzy yang menggambarkan derajat yang mana data input match dengan fungsi keanggotaan. Nilai data fuzzy adalah sebuah angka dan mempunyai range dari 0 sampai 63 khususnya didalam NLX220. Gambar 7 sebagai contoh evaluasi variabel fuzzy.



Gambar 3.7 Fuzifikasi Temperatur Input.

3.3.4 Aturan Fuzzy (Fuzzy Rule)

Rule terdiri dari satu atau lebih variabel fuzzy dan sebuah nilai aksi output. Rule digunakan untuk menceritakan bagaimana kontroler merespond perubahan

data input. Dalam sebuah contoh, kedua rule berisi dua variabel fuzzy. Rule dimasukkan ke sistem pengembangan INSIGHT dengan format sebagai berikut:

Output -5 If Velocity is Fast and Acceleration is Positive

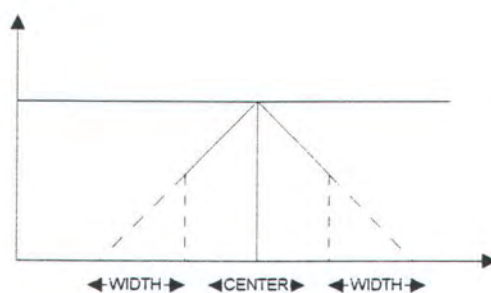
Output +5 If Velocity is Little- Slow and Acceleration is Zero.

Dalam rule pertama, variabel fuzzy pertama adalah 'Velocity is Fast' dan variabel fuzzy kedua adalah 'Acceleration is Positive.' Aksi '+5' atau '-5' adalah nilai angka yang dapat dipakai pada output untuk memperlambat atau mempercepat motor. Dalam contoh dibawah, tanda + digunakan untuk menunjukkan bahwa output tersebut dapat bertambah atau berkurang jika menggunakan mode akumulasi.

Ada beberapa metode untuk mengevaluasi rule logika fuzzy. NLX220 mengevaluasi rule menggunakan teknik dua step MAX dari MIN. Step pertama (MIN), semua nilai untuk variabel fuzzy dalam rule dibandingkan dan nilai terendah menggambarkan rule atau sebagai pemenang. Pada step kedua (MAX), nilai untuk rule dibandingkan dan rule dengan nilai tertinggi yang menjadi pemenang. Cara atau metode fungsi keanggotaan, variabel fuzzy dan rule-rule didefinisikan dan diorganisasi tergantung atas keperluan aplikasi. Sifat fisik sistem yang dikendalikan harus benar-benar dimengerti sebelum memasukkan model fuzzy. Namun dengan banyak pengalaman maka memasukkan sebuah model akan sama persis dengan proses.

3.3.5 Floating Membership Function

Rancangan unik dari NLX220 adalah fungsi keanggotaan mengapung. Seperti ditunjukkan gambar 3.8 dibawah, fungsi keanggotaan mengapung mempunyai nilai Center (pusat) dan Width (tepi) yang bervariasi secara dinamik. Fungsi keanggotaan biasa maka nilai center dan width adalah tetap (fixed) yang disimpan didalam memory. Fungsi keanggotaan mengapung tersebut nilainya berasal dari suatu input atau output.



Gambar 3.8 Fungsi Keanggotaan Mengapung.

Suatu fungsi keanggotaan dapat dipilih sebagai floating untuk memasukkan rancangan. Fungsi keanggotaan mengapung akan berubah nilai center dan nilai widthnya selagi data dari nilai input atau outputnya berubah. Sebagai contoh, dua variabel fuzzy dengan fungsi keanggotaannya yang menerangkan sepiintas lalu dant didefinisikan secara konvensional dan menggunakan rule sebagai berikut:

In1 is small (0,25, Symetrical Inclusive)

In2 is small (0.25, Symetrical Inclusive)

Dimana angka pertama adalah nol menunjukkan fungsi keanggotaan center dan yang kedua, 25 adalah fungsi keanggotaan width. Dua variabel dapat digabungkan menjadi sebuah rule sebagai berikut:

and IN2 is small Output +1 If IN1 is small

Dimana variabel fuzzy 'IN1 is small' membandingkan input IN1 terhadap fungsi keanggotaan konvensional 'small'. Fungsi keanggotaan mengapung dapat memberi penjelasan yang sama namun lebih ringkas dengan variabel fuzzy dan rule sebagai berikut:

IN1 is small-difference (IN2,25 Symetrical Exclusive)

Output +1 if IN1 is small-difference

Dalam variabel fuzzy, fungsi keanggotaan center small-difference didefinisikan oleh nilai IN2 yang disimpan dalam latch input. Dalam Fuzzifikasi, sebuah input dikurangkan dari fungsi keanggotaan center dan hasil dirubah untuk mengukur nilai absolut seberapa dekat match nilai centernya. Jika Fuzzifiernya sebuah fungsi keanggotaan floating center maka ia akan mengurangkan satu input dari yang lainnya. Fungsi keanggotaan mengapung memungkinkan menggunakan variabel fuzzy secara langsung mengukur perbedaan antara dua input. Teknik ini dapat digunakan untuk mengkalibrasi perubahan sebuah sensor waktu. Nilai sensor yang tetap dibandingkan terhadap himpunan tegangan Rule-rule kalibrasi mengecek derajat ketidak setaraan dan menyimpan nilai koreksi di sebuah latch output. Jika input dalam keadaan kalibrasi, center tersebut akan match dan nilai koreksinya nol.

Ketidak setaraan yang besar akan menyimpan koreksi yang besar. Koreksi digunakan untuk mengatur fungsi keanggotaan floating center dalam rule-rule yang memproses data yang dirasakan. Fungsi keanggotaan floating dapat digabungkan atau dikombinasi dengan nilai output aksi floating untuk memperoleh derivative dari sebuah nilai input. Rule dapat mereferensi sebuah input selagi nilai aksi floating menyebabkan ia dilewatkan secara langsung ke latch output. Selama sampling input berikutnya, nilai latch output memilih nilai fungsi keanggotaan center, yang mempengaruhi pengurangan nilai input sebelumnya dari nilai yang berlangsung (current value). Perbedaan atau selisih itu dibagi-bagi melalui interval sampling adalah merupakan nilai derivative yang dapat direferensikan dalam sebuah rule. Sebagai contoh menggunakan sebuah nilai aksi/input yang akan mengukur percepatan motor. Sebuah rule yang menyimpan sebuah input menjadi latch output dapat ditulis sebagai berikut:

VALUE- TO = IN1 If IN1 is MUST-WIN (0,0 Right Inclusive)

Rule mereferensi IN1 sebagai nilai aksi. Fungsi keanggotaan MUST-WIN adalah tipe Right Exclusive yang dimulai dari nol sedemikian rupa dengan tidak memandang nilai IN1, rule harus menang (win) dan nilai IN1 disimpan dioutput latch. Rule kedua mengkalkulasi derivative dan mengatur output yang menggerakkan motor.

ACCELL + If IN1 is value-T1 (value-T0,25 Symetrically inclusive)

Rule menentukan apakah nilai input T1 adalah dalam 25 dari nilainya pada T0. Dalam aplikasi nyata, ada fungsi keanggotaan lain yang menentukan polaritas derivative dan rule-rule lain untuk mengcover pengaturan yang lebih besar untuk variasi yang lebih besar. Contoh diatas adalah fungsi keanggotaan floating yang langsung (strightfoward). Dalam aplikasi nyata, fungsi keanggotaan floating dipakai secara intensip untuk menghemat memory karena hanya menggunakan beberapa variabel fuzzy dan rule-rule mendeteksi perbedaan antara input-input dibanding yang dilakukan fungsi konvensional.

3.4 Operasi NLX220

Pemrosesan data meliputi beberapa langkah. Pertama, data analog yang disampel dirubah menjadi digital dan ditahan (latched). Berikutnya Fuzzifier membandingkan isi input latch dengan variabel fuzzy untuk menemukan nilai variabel fuzzy. Fuzzifier juga melakukan kalkulasi MAX-of-MIN untuk menentukan rule yang menang. Akhirnya, Defuzzifier menentukan nilai aksi rule pemenang dan menahannya untuk mengkonversi menjadi output analog atau feedback internal.

3.4.1 Fuzzifier

Fuzzifier membandingkan data output yang ditahan dengan fungsi keanggotaan untuk menghitung nilai variabel fuzzy.

Ketika hitungan MIN telah dilakukan terhadap semua variabel fuzzy dalam sebuah rule, nilai yang mewakili rule disimpan. Ketika hitungan MAX telah dilakukan terhadap semua rule yang mereferensi sebuah output maka nilai aksi rule pemenang dilewatkan ke Defuzzier.

3.4.2 Pembaharuan Data Output

Rule-rule dievaluasi agar mereka dimasukkan. Output-output dapat digunakan sebagai referensi secara berulang-ulang dalam sebuah himpunan rule. Ketika rule atau sekelompok rule mempengaruhi output yang telah dievaluasi dan rule berikutnya yang dimasukkan mereferensi output yang lain, maka kompiller secara otomatis menyisipkan kode untuk Rule terakhir (Last Rule) menyebabkan output yang dilatch diperbaharui oleh nilai aksi rule pemenang. Data yang telah dilatch yang ada segera sebagai feedback. Jika, setelah pemrosesan rule mempengaruhi output-output yang lain, processor menghitung rule atau sekelompok rule yang lain yang mereferensi output sebelumnya, kemudian hal ini akan memperbaharui data latch output lagi.

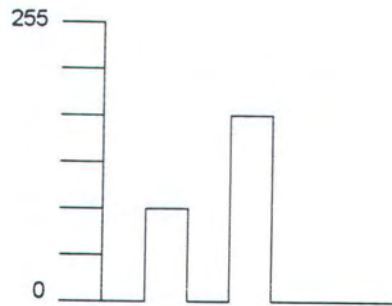
Sebuah data latch output kemudian dapat di perbaharui (di update) sebanyak waktu selama siklus pemrosesan selagi ada sekelompok rule yang terpisah yang mereferensinya. Seperti yang telah disinggung sebelumnya, penyamplingan input adalah kontinyu. Nilai output analog juga di update secara kontinyu.

Selama siklus pemrosesan, variabel fuzzy dapat menggunakan sebuah sampel data dari siklus sampel sebelumnya atau dari siklus yang sedang berlangsung (current) tergantung dimana letak siklus sampling relatif terhadap siklus pemrosesan. Akan ada lebih dari sekelompok rule yang mereferensi input dan output yang sama, kemudian nilai output dapat berubah lebih dari sekali selama siklus pemrosesan berdasarkan data input yang berbeda.

3.4.3 Defuzzifier

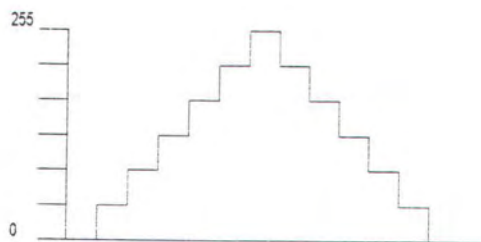
Nilai aksi rule pemenang dan mode data dilakukan ke blok Defuzzifier. Data digital dari Defuzzifier di latch dan dirubah menjadi analog untuk mendrive output atau di loop kembali secara internal (looped back internally). Jika semua kelompok rule mereferensi sebuah output yang mengevaluasi terhadap nol (zero), maka output tidak akan berubah nilainya. Jika lebih dari satu rule mengevaluasi terhadap nilai bukan nol tertinggi yang sama, maka rule yang pertama masuk yang akan **menang** dan aksinya akan menentukan output.

Defuzzifikasi menyebabkan nilai aksi rule pemenang mendrive (menggerakkan) sebuah output. Ada dua metode defuzzifikasi, Immediate dan Accumulate. Dua mode tersebut terlihat pada gambar 3.9 dan 3.10 dalam menseleksi rule. Fungsi mode Immediate seperti susunan tabel, dimana nilai aksi yang ditunjuk menjadi rule pemenang selama masuk dipakai menjadi output. Defuzzifikasi Immediate sangat berguna jika nilai output harus absolut.



Gambar 3.9 Defuzzifikasi Immediate.

Mode Accumulate, Mode Accumulate menambah atau mengurangi output yang ada oleh nilai aksi rule pemenang. Output adalah nilai aksi yang sedang berlangsung dan tergantung pada nilai aksi output sebelumnya. Defuzzifikasi accumulate dapat digunakan untuk perubahan halus pada output ketika sistem yang dikendalikan mendekati dekat titik set point yang diinginkan. Hal ini juga berguna untuk fungsi-fungsi waktu.



Gambar 3.10. Defuzzifikasi accumulate

3.5 Organisasi memory

NLX 220 berisi 256 byte memory OTP (One Time Program) untuk aplikasi penyimpanan parameter. 32 byte terakhir menyimpan nilai fungsi keanggotaan tetap (fixed member function) Center dan Width. Sisanya 224 byte diorganisasikan sebagai satu atau lebih rule dengan satu atau lebih variabel fuzzy per rule. Setiap rule memerlukan dua byte, plus dua byte tambahan untuk setiap variabel fuzzy dalam rule itu. Sebuah rule berisi lima variabel fuzzy untuk contoh akan menggunakan 12 byte. Memory diorganisasikan menjadi tiga seksi yang didefinisikan sebagai penyimpan variabel fuzzy/rule, Center, dan Width.

Tabel 3.1 Organisasi Memory

Alamat Desimal	Alamat Hexa	Fungsi
0	00	Rule
-----	-----	-----
223	DF	Rule
224	ED	Center
-----	-----	-----
239	EF	Center
240	FO	Width
-----	-----	-----
255	FF	Width

Rule diorganisasikan sebagai lebih kelompok variable fuzzy. Setiap variabel fuzzy terdiri dari dua byte, seperti dijelaskan dalam tabel 3.2 dan 3.3.

Byte pertama disimpan pada alamat genap dan yang kedua pada alamat ganjil. Byte dibagi menjadi bidang yang mengendalikan bagaimana data diproses. Tiga bit LSB untuk byte genap mendefinisikan tipe fungsi keanggotaan atau apakah variabel fuzzy sebelumnya adalah rule terakhir atau variabel fuzzy terakhir dari rule terakhir yang mereferensi output. Ketika bagian LSB memilih tipe fungsi keanggotaan, lima bit MSB dibagi menjadi tiga bagian bit yang memilih sumber satu input dari empat pin input atau latch output. Dua sisa bit MSB mendefinisikan Center dan Width dari fungsi keanggotaan floating atau fixed. Tipe kode signal variabel fuzzy terakhir (001) adalah variabel fuzzy terakhir dari rule yang telah diproses. Ketika hal ini terjadi, hanya dua bit MSB dari lima bagian bit yang digunakan. Bit MSB memilih apakah nilai aksi yang berasal dari lokasi memory fixed atau dari latch I/O. Bit MSB berikutnya memilih mode output apakah Immediate atau Accumulate. Kode (000) menunjukkan variabel fuzzy terakhir dari Rule terakhir. Dua bit diatas bagian pemilihan tipe digunakan untuk memilih output. Byte kedua selalu terjadi pada alamat ganjil dan berisi alamat Center dan alamat Width, bagian index jika byte sebelumnya dipilih sebagai tipe fungsi keanggotaan dan nilai Center atau Width Fixed. Jika salah satu baik itu Center atau Width dipilih sebagai floating, maka mereka mengambil byte ganjil digunakan untuk memilih input atau output.

Tabel 3.2 Organisasi byte genap (Command byte) dalam NLX220

7	6	5	4	3	2	1	0
WF	CF	I/O CONT	I/O SELECT		TYPE 2-7		
AF	MODE				TYPE 1		
AF	MODE		OUTPUT SELECT		TYPE 0		
Type		<u>210</u>					
		000	Last term of Last Rule of given output				
		001	Last Term of Current Rule				
		010	MF, Symmetrical Inclusive				
		011	MF, Symmetrical Exclusive				
		100	MF, Left Inclusive				
		101	MF, Left Exclusive				
		110	MF, Right Inclusive				
		111	MF, Right Exclusive				
I/O Select		<u>43</u>					
		00	I/O Port 0 as input				
		01	I/O Port 1 as input				
		10	I/O Port 2 as input				
		11	I/O Port 3 as input				
I/O Control		<u>5</u>					
		0	Select from inputs				
		1	Select from outputs				
Mode		<u>6</u>					
		0	Immediate, Output equals Action				
		1	Accumulate, Output equals current output plus two's complement action (-128 to 128)				
AF		<u>7</u>					
		0	Select Action from select byte (fixed)				
		1	Select Action from I/O via select byte (float)				
Output Select		<u>43</u>					
		00	ACTION from current RULE set to Output 0				
		01	ACTION from current RULE set to Output 1				
		10	ACTION from current RULE set to Output 2				
		11	ACTION from current RULE set to Output 3				
CF		<u>6</u>					
		0	Select Center from Memory via Select byte (fixed)				
		1	Select Center from I/O via select byte (float)				
WF		<u>7</u>					
		0	Select Width from Memory via Select byte (fixed)				
		1	Select Width from I/O via Select byte (fixed)				

Tabel 3.3 Organisasi Byte Ganjil (Select Byte) dalam NLX220

7	6	5	4	3	2	1	0	TYPE = 2 - 7 CF or WF = 0 (FIXED) TYPE = 2 - 7 CF or WF = 1 (FLOAT) TYPE = 0 - 1 AF = 0 (FIXED) TYPE = 0 - 1 AF = 1 (FLOAT)
CENTER SELECT				WIDTH SELECT				
	I/O CONT	I/O SELECT CENTER		I/O CONT	I/O SELECT WIDTH			
ACTION								
				I/O CONT	I/O SELECT ACTION			
Width Select		(3:0)	Used as Address index (E0-EF) for Fixed 6-bit WIDTH value when type = 2 - 7 and WF = 0					
Center Select		(7:4)	Used as Address index (F0-FF) for Fixed 8-bit CENTER value when type = 2 - 7 and CF = 0					
I/O Select Width		10						
		00	I/O Port 0 as Width (Type = 2 - 7 and WF = 1)					
		01	I/O Port 1 as Width (Type = 2 - 7 and WF = 1)					
		10	I/O Port 2 as Width (Type = 2 - 7 and WF = 1)					
		11	I/O Port 3 as Width (Type = 2 - 7 and WF = 1)					
I/O Control		2						
		0	Select from Inputs (Type = 2 - 7 and WF = 1)					
		1	Select from Outputs (Type = 2 - 7 and WF = 1)					
I/O Select Center		54						
		00	I/O Port 0 as Input (Type = 2 - 7 and CF = 1)					
		01	I/O Port 1 as Input (Type = 2 - 7 and CF = 1)					
		10	I/O Port 2 as Input (Type = 2 - 7 and CF = 1)					
		11	I/O Port 3 as Input (Type = 2 - 7 and CF = 1)					
I/ O Control		6						
		0	Select from Inputs (Type = 2 - 7 and CF = 1)					
		1	Select from Outputs (Type = 2 - 7 and CF = 1)					
ACTION		7:0	8-bit action value to be applied to an output due to winning Last Term of a Rule (Type = 1) or Last Term of last Rule of given output (Type = 0), and AF = 0 (Fixed)					
I/O Select Action		10						
		00	I/O Port 0 as Action (Type = 1 - 0 and AF = 1)					
		01	I/O Port 1 as Action (Type = 1 - 0 and AF = 1)					
		10	I/O Port 2 as Action (Type = 1 - 0 and AF = 1)					
		11	I/O Port 3 as Action (Type = 1 - 0 and AF = 1)					
I/O Control		2						
		0	Select from Inputs (Type = 1 - 0 and AF = 1)					
		1	Select from Outputs (Type = 1 - 0 and AF = 1)					

3.6 Pewaktuan (Timing)

Gambar 3.11 menunjukkan pewaktuan NLX 220. Ada tiga blok arsitek untuk pewaktuan meliputi pemultiplekan konverter A/D input, kontroler fuzzy dan pemultiplekan konverter D/A output. Kecepatan pemrosesan adalah fungsi kecepatan clock dan banyaknya clock (1024) yang diperlukan untuk penyamplingan data secara komplit dan siklus pemrosesan. Kecepatan maksimum clock adalah 10 Mhz dan minimum 1 MHz.

3.6.1 Pewaktuan Operasi

Reset.

Ketika pin RESET aktif, semua latch dihapus, output digital adalah dalam logika low dan output analog bertahan pada levelnya terutama pada saat reset. Jika RESET aktif untuk seratus clock atau lebih, input analog akan nol ketika sampling dimulai lagi. Jika RESET aktif untuk kurang dari seratus clock ada beberapa sisa data yang disampel terakhir yang masih ada pada input analog ketika penyamplingan dimulai lagi. Ketika RESET tidak aktif, maka penyamplingan input dimulai lagi selama 1024 clock.

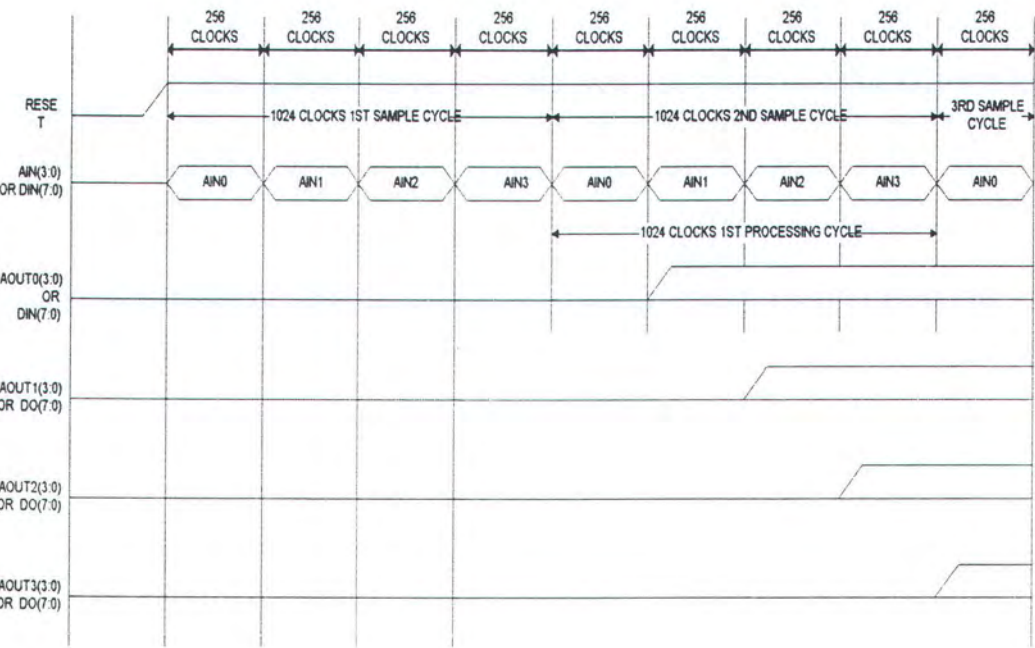
Konversi Input

Nilai analog input dirubah menjadi data digital dan dilatch secara internal dalam periode setiap 256 clock. Jumlah total 1024 clock yang diperlukan untuk mengkonversi empat input setiap pengulangan proses konversi.

Pada clock maksimum, kecepatan penyamplingan untuk setiap input adalah 10 Khz atau 100 mikrosecond.

3.6.2 Pewaktuan Kontroler

Siklus pemrosesan 1024 clock pertama mulai setelah siklus konversi input yang pertama komplit. Siklus pemrosesan terdiri 1024 clock dengan tidak memandang banyaknya variable fuzzy dan rule yang dipakai. Evaluasi variable fuzzy dan rule masing-masing memerlukan empat clock. Sebagai contoh, rule dengan dua variable memerlukan 12 clock untuk pemrosesan. Selama siklus pemrosesan baik variable fuzzy atau rule diproses setiap empat clock, kecuali 64 clock terakhir pada akhir siklus pemrosesan.



Gambar 3.11. Pewaktuan I/O

Ketika data berada dalam latch output akan diloopback secara internal sebagai input, maka mereka akan mengalami kelambatan sebagai input analog sebesar 1024 clock dari siklus sampling yang pertama. Setelah itu, pada saat latch output di update selama pemrosesan maka data feedback digunakan sebagai input.

3.6.3 Pewaktuan Output

Output di update pada batas 256 byte setelah pemrosesan dimulai seperti terlihat dalam gambar 3.11. Setiap pin output di update sekali setiap 1024 clock. Pewaktuan update output sangat tidak berubah. Latch output di update secara immediate setelah evaluasi rule yang relevan lengkap.

3.6.4 Operasi Prescale

Perangkat ini berisi sesuatu yang dapat dibebani, yaitu suatu counter prescale 8-bit yang memungkinkan perangkat ini menjadi inaktif untuk periode tertentu. Rancangan ini dapat digunakan pada berbagai kecepatan sampling dan pemrosesan. Lokasi terakhir dalam memory, yang dalam keadaan normal menyimpan data width dari fungsi keanggotaan fixed, dapat menyimpan suatu nilai yang dibebankan ke counter. Pin PRESCALE memilih operasi normal atau operasi prescale. Dalam mode prescale, kontroler inaktif untuk periode 1024 clock setelah counter ditambah. Ketika counter berjalan, kontroler aktif untuk periode tunggal 1024 clock untuk melakukan komputasi fuzzy dan counter dibebani lagi.

Dalam mengatur scale dengan memasukkan bilangan komplemen dari interval 1024 clock, yang diharapkan diantara siklus sampling dan pemrosesan. Sebagai contoh, dipilih penyamplingan yang dipisahkan dengan 2(dua) interval, berarti memasukkan bilangan komplemennya adalah FD. Interval-interval prescale dimasukkan pada saat kompilasi desain file. Pin dapat dikondisikan low ketika tidak digunakan atau dikondisikan ke pin READY untuk operasi prescale kontinyu. Logika eksternal juga dapat digunakan untuk mengoperasikan pin yang memungkinkan fungsi prescale dipertahankan atau tidak selama periode operasi yang berbeda.

3.6.5 Mode Inaktif

Konsumsi power dapat diperkecil dari mode operasi ke mode standby yaitu dengan mempertahankan pin clock high. Menghentikan clock berarti menunda pemrosesan dan membiarkan output pada kondisi setting terakhir. Nilai output analog akan menjadi nol. Pemrosesan berlanjut lagi ketika clock memulai lagi. Sedang clock berhenti pada level logika satu.

BAB IV

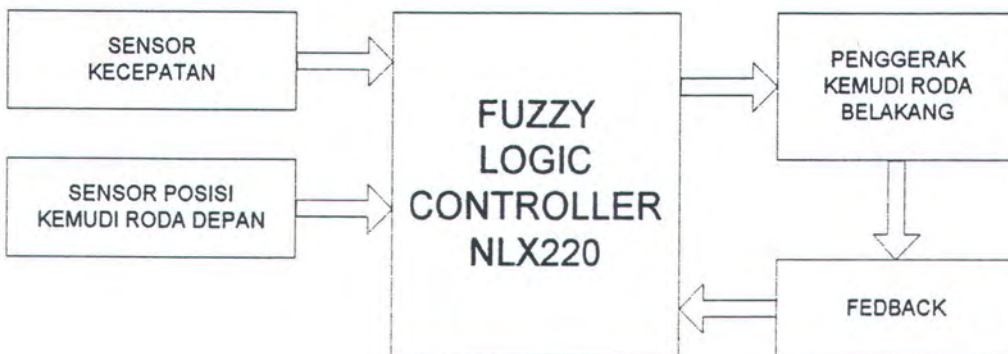
PERENCANAAN PERANGKAT KERAS

Dalam bab ini akan dibahas tentang perencanaan perangkat keras dari sistem yang akan dibuat. Untuk itu akan dilakukan langkah-langkah perancangan sebagai berikut :

1. Perancangan blok diagram sistem secara keseluruhan dan cara kerja.
2. Perancangan rangkaian elektronik dari setiap blok.

4.1 Perencanaan sistem

Blok diagram sistem yang akan dibuat diperlihatkan pada gambar .



Gambar 4.1 Blok diagram sistem

1. Sensor kecepatan berfungsi sebagai pengambil data untuk kecepatan kendaraan.
2. Sensor posisi kemudi roda depan berfungsi sebagai pengambil data untuk posisi sudut belok roda depan.

2. Sensor posisi kemudi roda depan berfungsi sebagai pengambil data untuk posisi sudut belok roda depan.
3. Fuzzy logic controller NLX220 berfungsi sebagai pemroses data logika fuzzy.
4. Penggerak kemudi roda belakang berfungsi sebagai output yang memberikan respon terhadap perubahan input.

4.2. Perencanaan perangkat

Seperti telah dijelaskan pada dasar teori bagian utama dari sistem yang akan dibuat adalah rangkaian tachometer, moving transducer, fuzzy logic sistem, dan penggerak kemudi.

4.2.1 Rangkaian sensor kecepatan

Kecepatan kendaraan sebanding dengan kecepatan putar roda, maka untuk mengetahui laju kendaraan cukup diambil data putaran roda. Pada tugas akhir ini digunakan frequency tachometer dimana sinyal output merupakan besaran tegangan yang variabel sesuai dengan kecepatan putaran roda. Dengan IC LM 2907 besaran frekuensi output dari sensor tersebut akan dikonversi kedalam besaran tegangan. Dengan memperhitungkan diameter roda kendaraan dapat dilakukan kalibrasi antara frekuensi dengan laju kendaraan
contoh : Roda kendaraan Timor dengan diameter 56 cm.

$$\begin{aligned}\text{Keliling roda kendaraan} &= \pi \cdot \text{diameter} \\ &= \pi \cdot 56 = 176\end{aligned}$$

Sehingga apabila roda berputar sekali maka kendaraan telah menempuh jarak 176 cm, jika dalam 1 detik roda berputar sekali berarti frekuensi putaran roda adalah 1 Hz. Berdasarkan perhitungan tersebut maka frekuensi 1 Hz setara dengan kecepatan 176 cm/detik, bila dikonversi dalam km/jam maka diperoleh:

$$1 \text{ jam} = 3600 \text{ detik}$$

$$\text{Dalam 1 jam ditempuh jarak } 176 \cdot 3600 = 633600 \text{ cm}$$

sehingga frekuensi 1 Hz sebanding dengan kecepatan 6,336 km/jam.

Untuk lebih meningkatkan ketelitian konversi maka pada roda ditempatkan 10 buah posisi sample, sehingga akan diperoleh perbandingan frekuensi 10 Hz setara dengan kecepatan kendaraan 6,336 km/jam.

Untuk mengubah frekuensi menjadi tegangan digunakan IC LM2907 buatan National Semiconductor. Pada IC LM2907 besarnya frekuensi yang masuk akan dikonversi menjadi tegangan DC. Untuk melakukan proses ini diperlukan 1 timing kapasitor, 1 output resistor, dan 1 filter kapasitor. Besarnya kenaikan tegangan output hasil konversi ditentukan oleh besarnya frekuensi, tegangan supply, dan komponen eksternal yang dipakai.

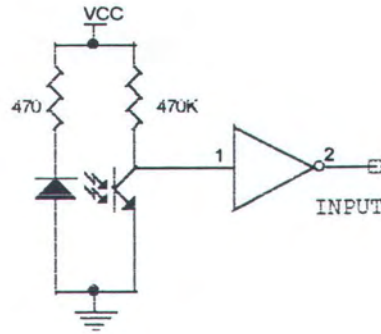
$$V_o = V_{cc} \cdot f_{in} \cdot R \cdot C \cdot K$$

Dengan mengambil asumsi bahwa kecepatan mobil maksimum adalah 180 km/jam maka f_{in} maksimum adalah 300 Hz, dan nilai typical K adalah 1. Harga typical kapasitor C adalah 0.05uF. Untuk mendapatkan range yang besar maka nilai V_o maksimum adalah 5 volt, sehingga besarnya R adalah

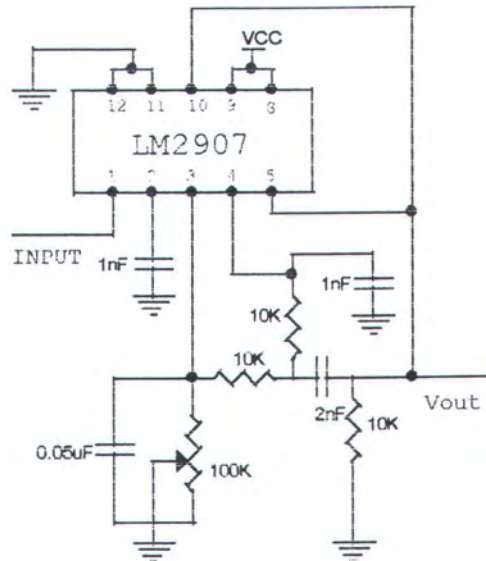
$$R = \frac{V_o}{V_{cc} \cdot f_{in} \cdot C}$$

$$R = \frac{5}{18 \cdot 300 \cdot 0.05 \cdot 10^{-6}}$$

$$R = 3.7 K\Omega$$



Gambar 4.2 Rangkaian opto isolator

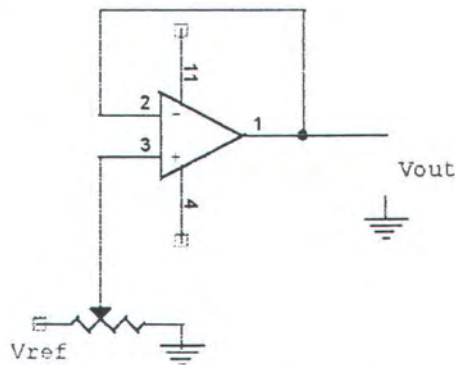


Gambar 4.3 Rangkaian Frekuensi to Voltage Converter

4.2.2 Sensor Posisi

Untuk mengetahui posisi putaran kemudi digunakan potensiometer, dimana besarnya sudut putar kemudi sebanding dengan besarnya resistansi antara terminal-terminal pada potensiometer. Dengan menambahkan sebuah Op-amp pada terminal

potensiometer maka diperoleh tegangan output yang sebanding dengan posisi putaran kemudi. Besarnya input maksimum untuk unit Fuzzy NLX220 adalah 5 volt maka besarnya nilai V_{ref} diset pada 5 volt.



Gambar 4.4 Rangkaian sensor posisi

4.2.3 Unit Fuzzy NLX220 system

Unit ini merupakan bagian utama yang akan memproses data dari transducer posisi dan tachometer. Pada bagian ini untuk input dibatasi pada range -0.6 sampai 4.7 volt dengan menggunakan diode dan zener. Pada bagian output langsung dihubungkan pada rangkaian driver motor DC.

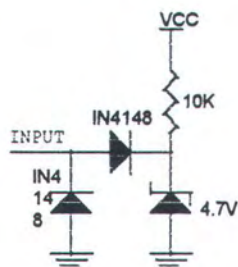
Frekuensi osilator dari NLX220 diset pada nilai yang tetap namun dapat juga diatur sesuai dengan yang diinginkan. Frekuensi osilator harus berada pada range 1 sampai 10 MHz. Dalam tugas akhir ini frekuensi osilator diset pada frekuensi 9 MHz. Penentuan besarnya frekuensi ini diatur dengan rangkaian RC seri, besarnya time konstan rangkaian RC seri adalah periode dari osilator. Besarnya nilai kapasitor tetap yaitu 22pF , untuk merubah besarnya time konstan dilakukan dengan merubah besarnya nilai R .

Besarnya nilai time konstan untuk rangkaian RC seri adalah

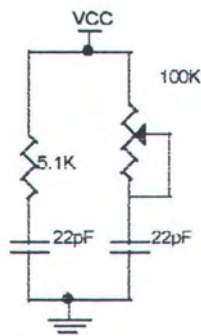
$$\tau = R \cdot C$$

$$R = \frac{1}{9 \cdot 10^6 \cdot 22 \cdot 10^{-12}} = 5.05 K\Omega$$

Chip NLX220 memiliki empat input dan empat output analog yang langsung dihubungkan dengan rangkaian yang lain.



Gambar 4.5 Rangkaian pembatas tegangan

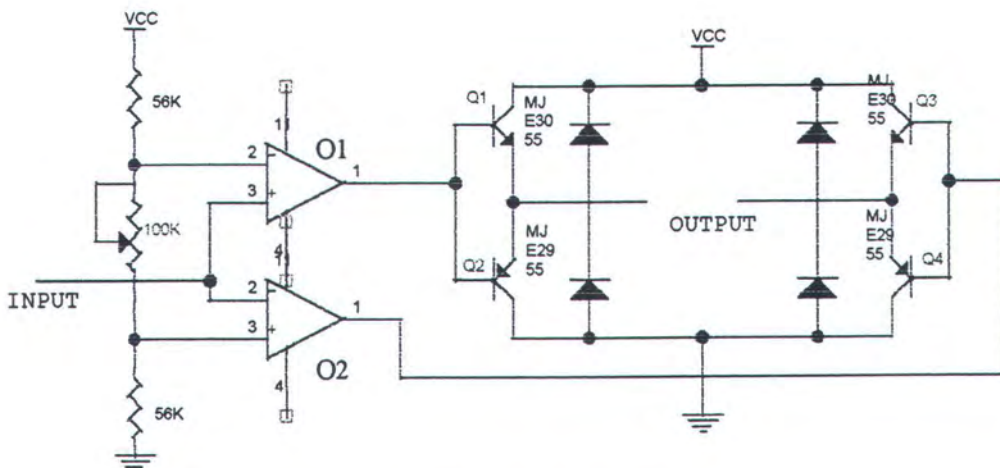


Gambar 4.6 Rangkaian resonator osilator

4.2.4 Rangkaian penggerak kemudi roda belakang

Pada perancangan sistem ini untuk penggerak roda belakang digunakan motor DC dengan reduksi putaran. Untuk mengatur agar motor bisa berfungsi sebagai kemudi digunakan rangkaian driver motor DC yang berfungsi untuk

mendrive motor agar dapat berputar dalam dua arah yang berbeda. Rangkaian driver motor terdiri atas 2 buah Op-Amp yang berfungsi sebagai komparator, dan 2 pasang transistor power.



Gambar 4.7 Rangkaian driver motor DC

Prinsip kerja rangkaian adalah, Op-Amp yang berfungsi sebagai komparator dan dipasang pada tegangan ambang tertentu. Rangkaian bekerja dalam 3 kondisi keadaan yaitu :

- Untuk tegangan input melebihi batas tegangan ambang tersebut maka output Op-Amp O1 adalah tegangan saturasi dan output Op-amp O2 adalah 0. Sehingga transistor Q1 dan Q4 aktif sedangkan transistor Q2 dan Q3 cut off.
- Untuk tegangan input berada pada batas tegangan ambang tersebut maka output Op-Amp O1 adalah 0 dan output Op-amp O2 adalah 0. Sehingga transistor Q1, Q2, Q3, Q4 cut off.

- Untuk tegangan input kurang dari batas tegangan ambang tersebut maka output Op-Amp O1 adalah 0 dan output Op-amp O2 adalah tegangan saturasi. Sehingga transistor Q1 dan Q4 cut off sedangkan transistor Q2 dan Q3 aktif.

Untuk mengubah arah putaran pada motor DC, dilakukan dengan memberikan tegangan input yang berada diluar batas tegangan ambang agar rangkaian bekerja pada kondisi pertama dan ketiga. Penggunaan tegangan ambang (histerisis) bertujuan untuk mencegah kesalahan arah putaran motor bila input mengalami ripple.

4.2.5 Rangkaian Display

Untuk mengetahui kondisi kendaraan dipasang rangkaian display 7 segment yang akan menampilkan kecepatan kendaraan dan sudut kemudi roda depan.

Untuk menampilkan digunakan IC ICL7107 yang merupakan AD converter 3 ½ digit. IC ini akan melakukan konversi terhadap beda potensial (differensial) antara dua pin inputnya.

BAB V

PERENCANAAN PERANGKAT LUNAK

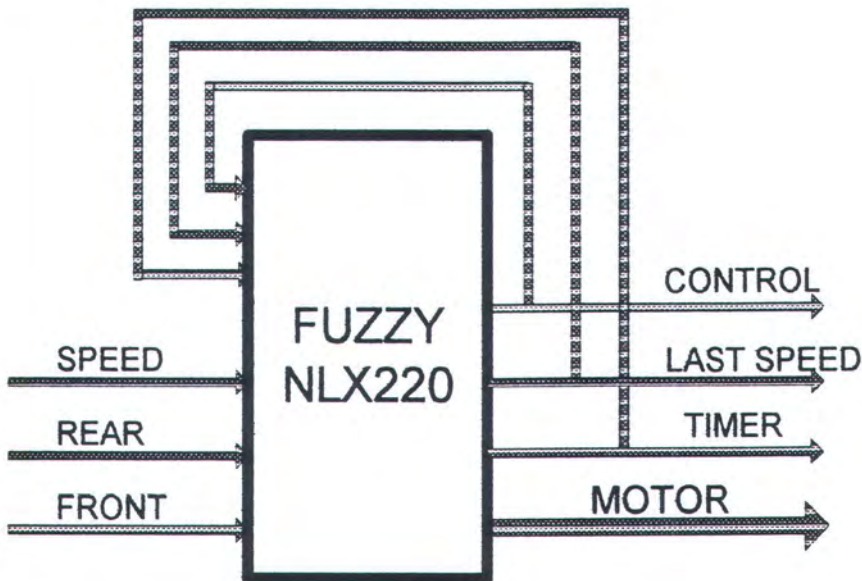
Software untuk NLX220 dirancang untuk mendeteksi beberapa kejadian yaitu:

- Besarnya kecepatan kendaraan
- Besarnya sudut putar kemudi
- Arah sudut putar kemudi
- Aksi yang harus dilakukan

Untuk itu digunakan pendekatan fungsi kontrol PID dimana input ke fuzzy controller adalah nilai kecepatan kendaraan, nilai dan arah sudut putar kemudi. Selanjutnya fuzzy akan mendeteksi nilai kedua besaran tersebut serta perubahannya. Besarnya aksi yang dilakukan oleh fuzzy controller merupakan respon yang besarnya sesuai dengan standard sistem kemudi yang dipakai acuan. Dalam tugas akhir ini digunakan sistem kemudi standar Honda dan sistem kemudi standard Mazda.

Dari kedua sistem kemudi tersebut diperoleh data bahwa sistem kemudi standard Honda mempunyai keunggulan pada kecepatan tinggi sedangkan sistem kemudi standard Mazda mempunyai keunggulan pada kecepatan rendah. Berdasarkan hal tersebut maka akan dibuat sebuah input mengambang (floating membership function) yang berfungsi untuk menyeleksi kondisi mana yang lebih baik bila kedua input memberikan memberikan nilai yang berbeda.

Blok diagram sistem diperlihatkan sebagai berikut:



Gambar 5.1 Diagram input output sistem

Sistem yang dibuat menggunakan beberapa input dan feedback.

Input untuk sistem adalah:

1. Speed, merupakan laju kecepatan
2. Rear, merupakan posisi dari sudut belok roda belakang
3. Front, merupakan posisi sudut belok roda depan

Data input speed merupakan besarnya laju kecepatan kendaraan. Data input rear merupakan besarnya sudut belok roda belakang. Data input front merupakan besarnya sudut belok roda depan. Data input untuk speed diambil dari output rangkaian frekuensi to voltage converter, sedangkan untuk rear dan front diambil dari rangkaian sensor posisi yang mendeteksi perubahan putaran sudut.

Feedback untuk sistem adalah:

1. Control
2. Lastspeed
3. Timer

Control merupakan variabel dari kondisi aksi yang diinginkan, besaran ini akan digunakan sebagai pembanding terhadap posisi sudut belok roda belakang. Lastspeed adalah kondisi kecepatan kendaraan terakhir, nilai ini digunakan untuk mendeteksi perubahan kecepatan yang terjadi dalam selang waktu tertentu yang diinginkan. Timer merupakan pewaktu yang kecepataannya disesuaikan dengan respon output yang diinginkan.

Output untuk sistem digunakan variabel Motor yang akan mengatur posisi roda belakang melalui rangkaian driver motor DC.

Untuk masing-masing input digunakan membership function sebagai berikut :

1. Speed

Speed is same (Lastspeed,2,Symetrical Inclusive)

Speed is smdrop (Lastspeed,2,Right Exclusive)

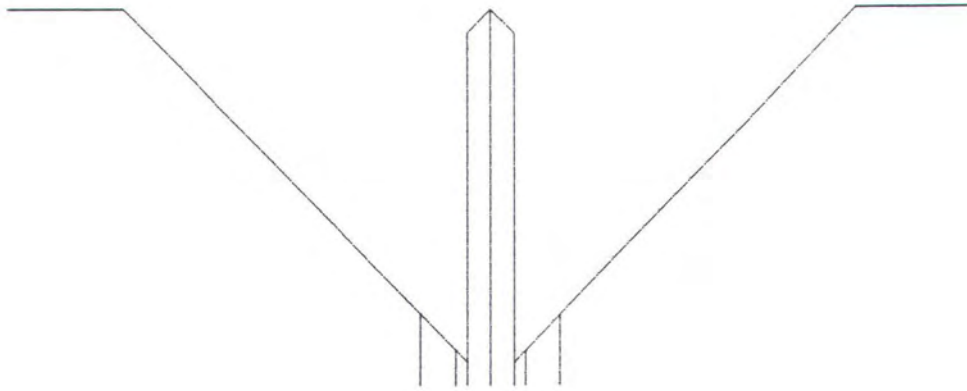
Speed is smrise (Lastspeed,2, Left Exclusive)

Speed is meddrop (Lastspeed,6, Right Exclusive)

Speed is medrise(Lastspeed,6, Left Exclusive)

Speed is lrgdrop (Lastspeed,12, Right Exclusive)

Speed is lgrise (Lastspeed,12, Left Exclusive)



Gambar 5.2 Membership function untuk Speed input

2. Front

Front is HigHgLef (106,44,Right Exclusive)

Front is NorHgLef (106,40,Right Exclusive)

Front is LowHgLef (106,36,Right Exclusive)

Front is NormLef (106,32,Right Exclusive)

Front is HigMdLef (106,28,Right Exclusive)

Front is NorMdLef (106,24,Right Exclusive)

Front is LowMdLef (106,20,Right Exclusive)

Front is PeakLef (106,16,Right Exclusive)

Front is HigSmLef (106,12,Right Exclusive)

Front is NorSmHgLef (106,8,Right Exclusive)

Front is LowSmLef (106,4,Right Exclusive)

Front is Foward (106,4,Symetrical Inclusive)

Front is HigHgRig (106,44,Left Exclusive)

Front is NorHgRig (106,40,Left Exclusive)

Front is LowHgRig (106,36,Left Exclusive)

Front is NormRig (106,32,Left Exclusive)

Front is HigMdRig (106,28,Left Exclusive)

Front is NorMdRig (106,24,Left Exclusive)

Front is LowMdRig (106,20,Left Exclusive)

Front is PeakRig (106,16,Left Exclusive)

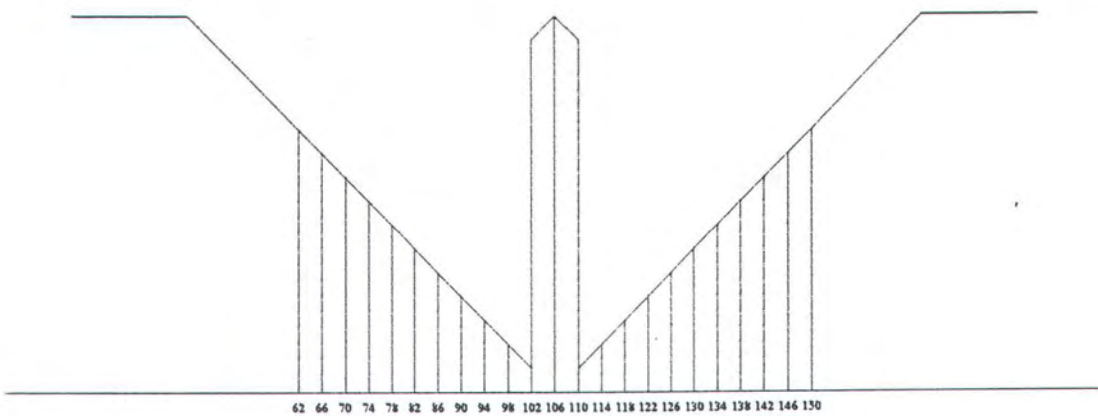
Front is HigSmRig (106,12,Left Exclusive)

Front is NorSmHgRig (106,8,Left Exclusive)

Front is LowSmRig (106,4,Left Exclusive)

Front is Right (106,4,Right Inclusive)

Front is Left (106,4, Left Inclusive)



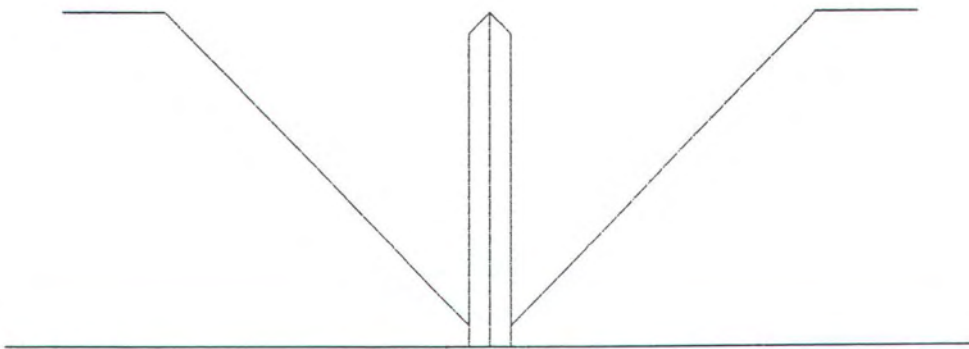
Gambar 5.3 Membership function untuk Front input

3. Control

Control is sama (rear,5,symmetrical inclusive)

Control is kurang (rear,5,Right Exclusive)

Control is kurang (rear,5,Left Exclusive)



Gambar 5.4 Membership function untuk Control

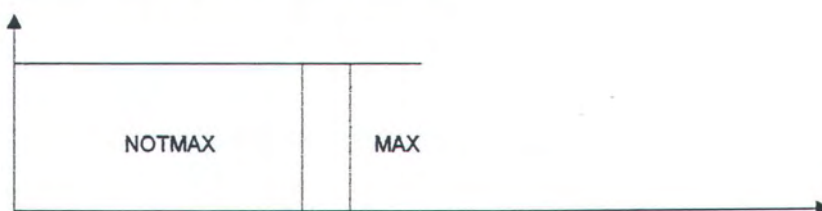
4. Timer

Timer is max(250,0,right inclusive)

Timer is notmax(249,0,left inclusive)

Timer is bebas(0,0,right inclusive)

Timer is zero(0,0,symmetrical inclusive)



Gambar 5.5 Membership function untuk Timer

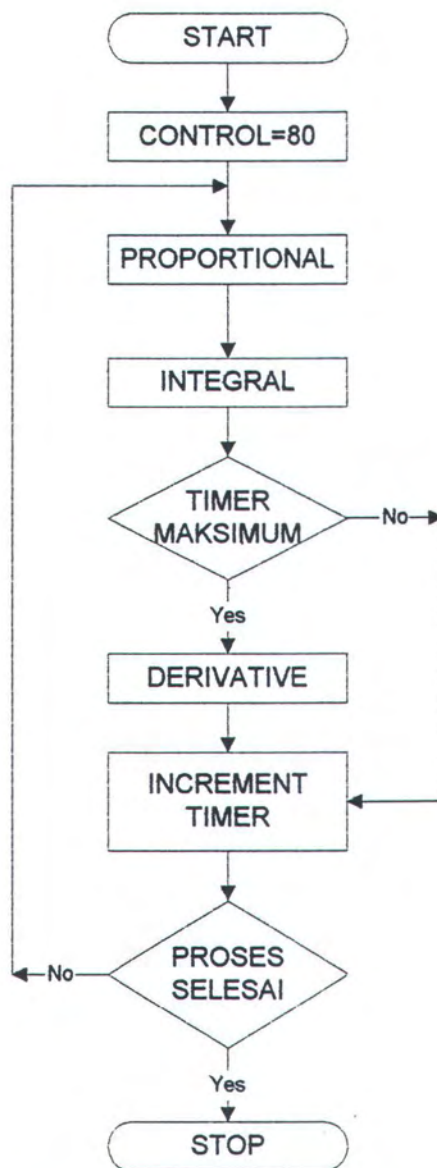
Langkah selanjutnya membuat sekelompok rule yang akan melakukan aksi sesuai dengan input dan perubahan input. Rule-rule ini akan dikelompokkan menjadi tiga bagian besar yaitu rule integral, proportional, dan derivative.

Processor NLX220 menggunakan sekelompok rule untuk setiap output, dari setiap kelompok tersebut pada saat diproses maka rule yang menang nilai aksinya akan mewakili kelompok tersebut dan keluar melalui outputnya. Jika satu atau beberapa kelompok rule menggunakan output yang sama maka harus digunakan rule pembatas (dummy rule) agar aksi rule pemenang dari kelompok pertama tidak hilang digantikan oleh aksi rule kelompok berikutnya. Pada perencanaan tugas akhir ini akan digunakan variabel timer sebagai variabel pembatas antar kelompok rule yang menggunakan output yang sama.

Dalam tugas akhir ini untuk menentukan besarnya nilai output digunakan dua mode yaitu Accumulate dan Immediate. Penggunaan mode Immediate dilakukan dengan melakukan fungsi pemetaan langsung setiap kondisi input, cara ini mewakili metode Proportional dalam kontrol PID. Penggunaan mode Accumulate dilakukan dengan melakukan penjumlahan dan pengurangan terhadap variabel control, fungsi penjumlahan dan pengurangan pada setiap perubahan input ini mewakili metode integral dalam kontrol PID. Metode diferensial diwakili dengan menggunakan feedback Lastspeed yang digunakan untuk mendapatkan nilai perubahan input kecepatan kendaraan. Untuk mengatur kecepatan respon dari peralatan digunakan variabel timer, penentuan timer

dilakukan hanya saat perancangan, setelah ditentukan besarnya kecepatan respon yang diinginkan variabel timer dibuat tetap.

Berikutnya adalah perencanaan bagan alir cara kerja rule.



Gambar 5.6 Diagram alir proses kerja rule

BAB VI

PENGUJIAN DAN PENGUKURAN

Sebelum sistem dijalankan, maka perlu dilaksanakan pengujian untuk tiap bagian sistemnya dan selanjutnya dilakukan kalibrasi dan pengukuran terhadap alat yang direncanakan.

6.1 Pengujian alat

Pengujian dilakukan untuk mengetahui alat bekerja dengan baik atau tidak, menguji fungsi-fungsi yang telah direncanakan serta mengetahui kinerjanya.

Pengujian dilakukan pada modul yang telah dibuat, yaitu:

Modul Frekuensi to Voltage Converter

Modul Fuzzy NLX220

Modul driver motor DC

Pengujian modul Frekuensi to Voltage Converter meliputi pengujian output tegangan dengan menggunakan frekuensi-frekuensi tertentu, dan mengukur tegangan yang keluar. Hasil pengukuran yang telah dikalibrasi untuk nilai output maksimal 5 volt ditunjukkan pada tabel 6.1.

Pengujian modul fuzzy NLX220 adalah dengan membuat rule-rule sederhana yang akan melewatkan sinyal input langsung ke outputnya.

Contoh rule sederhana untuk menguji modul adalah:

Input1 is win1(0,0,right inclusive)

Input2 is win2(0,0,right inclusive)

Input3 is win3(0,0,right inclusive)

Input4 is win4(0,0,right inclusive)

Rule yang digunakan adalah :

If input is win1 then output1=input1

If input is win2 then output2=input2

If input is win3 then output3=input3

If input is win4 then output4=input4

Dengan menggunakan kelompok rule tersebut maka pada setiap input diberi sumber tegangan dan outputnya diukur. Dari pengukuran diperoleh nilai yang sesuai, berarti rangkaian telah bekerja dengan baik.

6.2 Pengujian sistem secara lengkap

Setelah bagian per bagian dilakukan pengujian maka selanjutnya digabungkan dan dipakai pada sistem kemudi kendaraan. Pada kondisi awal rangkaian diset pada sudut 0 derajat kemudian diberi input. Apabila rangkaian bekerja baik pada input tersebut dan melakukan pergerakan kemudi yang tajam pada setiap perubahan input, maka proses selanjutnya adalah mengubah nilai setting awal pada nilai increment rule sesuai dengan yang dikehendaki.

Selanjutnya dilakukan pengujian dengan melakukan perubahan input pada kecepatan dan posisi kemudi secara bergantian dan bersamaan. Hasil pengukuran pengaruh sudut belok roda belakang terhadap kecepatan dan sudut kemudi ditunjukkan pada tabel 6.2 dan tabel 6.3.

Untuk mendapatkan data tentang kecepatan kendaraan dilakukan pengujian dan pengukuran terhadap rangkaian frekuensi to voltage converter.

Tabel 6.1 Perubahan tegangan terhadap frekuensi

Laju kendaraan (km/jam)	Frekuensi input (Hz)	Tegangan output (volt)	Perhitungan (volt)	error
6	10	0.2	0.27	26%
12	20	0.48	0.54	11%
18	30	0.75	0.81	7%
24	40	1	1.08	7%
30	50	1.25	1.35	7%
36	60	1.5	1.62	7%
42	70	2	1.89	6%
48	80	2.25	2.16	4%
54	90	2.55	2.43	5%
60	100	2.8	2.70	4%
75	125	3.5	3.37	4%
90	150	3.9	4.050	4%
100	165	4.3	4.45	3%
105	175	4.6	4.72	3%
110	185	4.9	4.99	2%
120	200	5.3	5.40	2%

Pengujian selanjutnya adalah perubahan sudut belok roda belakang terhadap perubahan kecepatan. Berdasarkan standard Mazda maka perubaha sudut belok roda belakang sebanding dengan kecepatan kendaraan. Pada kecepatan normal sudut belok roda belakang tidak mengalami perubahan.

Tabel 6.2 Perubahan sudut belok terhadap kecepatan

Kecepatan kendaraan (km/jam)	Sudut belok (derajat)	Perhitungan (derajat)	error
6	-25	-25	0%
12	-20	-22	0.6%
18	-17	-20	0.8%
24	-12	-15	0.8%
30	-8	-12	1.1%
36	-1	0	0.2%
42	5	8	0.8%
48	8	12	1.1%
54	12	17	1.4%
60	18	20	0.6%
75	22	25	0.8%
90	25	25	0%
105	25	25	0%
120	25	25	0%

Pengujian selanjutnya adalah perubahan sudut belok roda belakang terhadap perubahan sudut belok roda depan. Berdasarkan standard Honda maka perubahan sudut belok roda belakang pada sudut belok roda depan yang kecil akan menghasilkan sudut belok yang searah, sebaliknya perubahan sudut belok roda belakang pada sudut belok roda depan yang besar akan menghasilkan sudut belok yang berlawanan arah. Pada sudut belok roda depan tertentu, maka sudut belok roda belakang akan berada pada posisi sudut nol derajat.

Tabel 6.3 Perubahan sudut belok terhadap posisi kemudi

Sudut belok roda depan (derajat)	Sudut belok roda belakang (derajat)	Perhitungan (derajat)	error
0	0	0	0%
5	5.5	7.5	0.6%
10	10	12.5	0.7%
15	6	7.5	0.5%
20	1	0	0.2%
25	-6	-7.5	0.5%
30	-10	-12.5	0.7%
35	-15	-18	0.8%
40	-22	-25	0.8

Tabel 6.4 Perubahan sudut belok roda belakang untuk input kecepatan rendah

Sudut belok roda depan (derajat)	Kecepatan kendaraan (km/jam)	Sudut belok roda belakang (derajat)
40	2.5	-25
40	5	-25
40	10	-20
40	15	-16
40	20	-12
40	25	-8
40	30	-5
40	35	2
40	40	8

Tabel 6.5 Perubahan sudut belok roda belakang untuk input kecepatan tinggi

Sudut belok roda depan (derajat)	Kecepatan kendaraan (km/jam)	Sudut belok roda belakang (derajat)
40	35	2
40	40	8
40	45	14
40	50	16
40	55	20
40	60	22
40	65	25
40	70	25
40	75	25

6.3 Faktor adaptif

Pada tabel 6.4 dan tabel 6.5 diatas adalah hasil pengukuran untuk menentukan faktor adaptif dari sitem kontrol. Pada pengujian ini simpangan (sudut belok) roda depan dibuat maksimum. Pengujian faktor adaptif dilakukan untuk melihat kemampuan fuzzy controller dalam memilih standard gerakan roda belakang yang paling baik untuk suatu kondisi yang bervariasi secara acak.

Pada saat kecepatan rendah pergerakan kemudi roda belakang sepenuhnya mengikuti standard Honda. Pada pengujian selanjutnya kecepatan ditambah terus, sistem kontrol akan mengevaluasi perubahan kecepatan ini untuk menentukan respon outputnya. Pada saat kecepatan mulai mendekati kecepatan normal (sekitar 35 km/jam) sistem kontrol akan mulai mengabaikan standar Honda dan berpindah menggunakan standar Mazda untuk menentukan respon outputnya. Selanjutnya pada kecepatan tinggi gerakan roda belakang sepenuhnya mengikuti standard Mazda.

BAB VII

PENUTUP

7.1 Kesimpulan

Dari pembuatan tugas akhir ini diperoleh beberapa kesimpulan penting yaitu:

1. Suatu proses pengontrol sistem kemudi 4 roda dengan Fuzzy Logic Controller NLX220 bisa menyediakan suatu operasi pengontrolan yang baik, optimal biaya murah, dan dengan ukuran yang kecil.
2. Proses pengontrolan sistem kemudi 4 roda dengan NLX220 berhasil baik pada daerah yang jauh dari sudut netral kemudi (rear kemudi sudut nol derajat).
3. Rangkaian frekuensi to voltage converter bisa mewakili fungsi speedometer pada kendaraan.
4. Kontroller dapat berfungsi dengan lebih baik bila ditunjang dengan penggunaan motor penggerak kemudi yang lebih presisi.
5. Penggunaan fuzzy microcontroller NLX220 ternyata mampu memecahkan permasalahan pada sistem kemudi 4 roda, sehingga fuzzy terbukti keandalannya.

7.2 Saran-saran

Untuk pengembangan lebih lanjut tugas akhir ini maka ada beberapa saran yang bisa berguna diantaranya:

1. Perlu digunakan beberapa variabel input yang lain untuk lebih mengoptimalkan kerja rangkaian, misalnya besarnya diferensial putaran roda.
2. Untuk mendapatkan hasil yang lebih presisi bisa digunakan motor servo sebagai penggerak kemudi.
3. Untuk mengontrol sistem kemudi 4 roda dapat pula dilakukan dengan mengambil input dari besarnya gaya sentrifugal yang bekerja pada kendaraan.
4. Untuk mendapatkan hasil yang optimal perlu dilakukan kalibrasi dari rangkaian frekuensi to voltage converter, sehingga bisa berfungsi sebagai speedometer.
5. Bila rangkaian dipasang pada kendaraan lain maka perlu dilakukan penyesuaian dari besarnya respon.
6. Untuk lebih meningkatkan keandalan sistem kontrol perlu ditambahkan rangkaian pengaman yang akan mengembalikan roda belakang ke posisi normal bila sistem kontrol mengalami kerusakan.

DAFTAR PUSTAKA

1. Malvino,Barmawi, PRINSIP-PRINSIP ELEKTRONIKA, jilid I, edisi ketiga, Eralangga,1986.
2. Malvino,Barmawi, PRINSIP-PRINSIP ELEKTRONIKA, jilid II, edisi ketiga, Eralangga,1986.
3. Millman dan Halkias, M.Barmawi, M.O. Tjia, ELEKTRONIKA TERPADU, jilid II, Erlangga, 1985.
4. Ogata,Katsuhiko,TEKNIK KONTROL AUTOMATIK, jilid I, diterjemahkan oleh Edi Leksono, Jakarta, Erlangga, 1991.
5. Robert F. Coughlin, Frederick F. Driscoll, PENGUAT OPERASIONAL DAN RANGAKAIAN TERPADU LINIER, edisi kedua, diterjemahkan oleh Herman Widodo Sumitro, Erlangga, Jakarta, 1992.
6. _____Yan, Jun, Michael Ryan, dan James Power, USING FUZZY LOGIC, Prentice Hall International, 1994.
7. _____,APROXIMATION OF PID CONTROLLER USING THE AL220 FUZZY CONTROLLER, Adaptive Logic inc.
8. _____,FUZZY LOGIC EDUCATION PROGRAM, Motorola.
9. _____,NLX220, Stand Alone Fuzzy Logic Controller Preliminary Data,Adaptive Logic inc.

BIODATA



Doris Erwantoro dilahirkan di Lumajang Jawa Timur pada tanggal 13 Juni 1974 dari pasangan Bapak Soedjana dan Ibu Sopeniati yang saat ini bertempat tinggal di Jalan Ikan Duyung 10 Malang dan merupakan anak keempat dari 4 bersaudara.

Riwayat Pendidikan penulis adalah sebagai berikut :

- Tahun 1986 : Lulus SD Kaliboto 5 Jatiroto
- Tahun 1989 : Lulus SMP 11 Malang
- Tahun 1992 : Lulus SMA 3 Malang
- Tahun 1992 : Diterima di Jurusan Teknik Elektro FTI-ITS dengan NRP 2292.100.081 melalui program UMPTN.

Pada bulan Agustsus 1998 mengikuti seminar dan ujian Tugas Akhir pada Bidang Studi Elektronika dan diharapkan dapat mengikuti wisuda pada bulan September 1998.

Selama berkuliah di Jurusan Teknik Elektro FTI-ITS penulis aktif dalam kegiatan kemahasiswaan antar lain :

- Koordinator Praktikum Elektronika tahun 1996-1997
- Asisten praktikum Rangkaian Listrik, Elektronika, dan Elektronika Lanjutan II di Laboratorium Bidang Studi Elektronika.

Rule-rule yang digunakan dalam perencanaan sistem ini adalah:

If Front is HigHgLef
Then control = 125
If Front is NorHgLef
Then control = 110
If Front is LowHgLef
Then control = 95
If Front is NormLef
Then control = 80
If Front is HigMdLef
Then control = 65
If Front is NorMdLef
Then control = 50
If Front is LowMdLef
Then control = 35
If Front is PeakLef
Then control = 20
If Front is HigSmLef
Then control = 35
If Front is NorSmHgLef
Then control = 50
If Front is LowSmLef
Then control = 65
If Front is HigHgRig
Then control = 35
If Front is NorHgRig
Then control = 50
If Front is LowHgRig
Then control = 65
If Front is NormRig
Then control = 80
If Front is HigMdRig
Then control = 95
If Front is NorMdRig
Then control = 110
If Front is LowMdRig
Then control = 125
If Front is PeakRig
Then control = 140
If Front is HigSmRig
Then control = 125
If Front is NorSmHgRig
Then control = 110

If Front is LowSmRig
Then control = 95
If Front is Forward
Then control = 80
If timer is max
Then timer = 1
If Front is Right and speed is lrgdrop
Then control + -10
If Front is Left and speed is lrgdrop
Then control + 10
If Front is Right and speed meddrop
Then control + -6
If Front is Left and speed is meddrop
Then control + 6
If Front is Right and speed is smdrop
Then control + -2
If Front is Left and speed is smdrop
Then control + 2
If Front is Right and speed is lgrise
Then control + 10
If Front is Left and speed is lgrise
Then control + -10
If Front is Right and speed medrise
Then control + 6
If Front is Left and speed is medrise
Then control + -6
If Front is Right and speed is smrise
Then control + 2
If Front is Left and speed is smrise
Then control + -2
If timer bebas
Then timer +20
If timer max
Then Lastspeed is speed
If control is lebih
Then motor = 0
If control is sama
Then motor = 128
If control is kurang
Then motor = 255